Distribution
Category UC-70

SAND87-1176
Unlimited Release
Printed February 1988

SAND--87-1176-Vol.3

VOLUME III

TI88 005816

Description of Ground Motion
Data Processing Codes

By

Michele L. Sanders
Ground Motion and Seismic Division
Sandia National Laboratories
Albuquerque, New Mexico 87185

Abstract

Data processing codes developed to process ground motion at the Nevada Test Site for the Weapons Test Seismic Investigations Project are used today as part of the program to process ground motion records for the Nevada Nuclear Waste Storage Investigations Project. The work contained in this report documents and lists these codes and verifies the "PSRV" code.

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

MASTER

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

Summary

This report, in three volumes, describes the ground motion data processing codes in detail. Volume I, which contains the program specifications for the scientific and engineering software, consists of the following programs: "SPECTRA" calculates the power density spectrum of digitized time histories from underground nuclear explosions (UNEs); "FILTER" designs optimum finite impulse response filters and then filters UNE ground motion data; and "PSRV" calculates the pseudo relative velocity response spectrum for UNE ground motions. Volume II contains the following program specifications for the auxiliary software. "ACCESS" allows the user to scan, update, delete, or add records contained in the Weapons Test Seismic Investigations data base. "SCAN" produces an inventory of the time histories contained in a large data file. "PLOT" plots the data files; "ROTATE" converts data in the horizontal plane into a UNE specific coordinate system; and "VECTOR" calculates the magnitude versus time of the 2-D (horizontal components) and 3-D (vertical and horizontal components) vectors. Volume III comprises the users' manuals for each of the programs described in Volumes I and II.

VOLUME III
USER MANUALS
TABLE OF CONTENTS

CHAPTER 9
"ACCESS" USER MANUAL

9.1. GENERAL INFORMATION	9-2
9.1.1. Summary.	9-2
9.1.2. Environment.	9-2
9.1.3. References.	9-2
9.2. APPLICATION	9-3
9.2.1. Description.	9-3
9.2.1.1. When to Use this Program.	9-3
9.2.1.2. Program Functions.	9-3
9.2.2. Operation.	9-3
9.2.3. Equipment.	9-3
9.2.4. Structure.	9-3
9.2.5. Performance.	9-4
9.2.6. Data Base.	9-4
9.2.6.1. The Event File.	9-4
9.2.6.2. The Station File.	9-4
9.2.6.3. The Canister File.	9-4
9.2.6.4. The Gauge File.	9-5
9.2.7. Inputs, Processing, and Outputs.	9-5
9.2.7.1. Inputs.	9-5
9.2.7.2. Interacting with the Program.	9-5
9.2.7.3. Outputs.	9-5
9.3. PROCEDURES AND REQUIREMENTS	9-7
9.3.1. Initiation.	9-7
9.3.1.1. Locating the Files Into the User File Space.	9-7
9.3.1.2. Compiling the Program.	9-7
9.3.1.3. Loading the Program Into Memory.	9-7
9.3.1.4. Executing the Program.	9-8
9.3.2. Input.	9-8
9.3.2.1. Requirements.	9-8
9.3.2.1.1. Frequency.	9-8
9.3.2.1.2. Origin.	9-8
9.3.2.1.3. Medium.	9-8
9.3.2.1.4. Restrictions.	9-8
9.3.2.2. Input Formats.	9-9
9.3.2.2.1. Length.	9-9
9.3.2.2.2. Format.	9-9
9.3.2.2.3. Labels.	9-9
9.3.2.2.4. Sequence.	9-9
9.3.2.2.5. Combination.	9-9

9.3.3. Sample Inputs.	9-10
9.3.3.1. Track Sheets.	9-10
9.3.3.2. Canister Logs.	9-10
9.3.3.3. Accelerometer Calibration Reports.	9-10
9.3.4. Output.	9-17
9.3.4.1. Use.	9-17
9.3.4.2. Frequency.	9-17
9.3.4.3. Medium.	9-17
9.3.4.4. Disposition.	9-17
9.3.4.5. Output.	9-18
9.3.5. Error and Recovery.	9-18
9.3.6. File Query.	9-18
9.3.6.1. Selecting a Data Base File.	9-18
9.3.6.2. Selecting a Data Base Function.	9-19
9.3.6.3. Data Base Functions.	9-19
9.3.6.3.1. Add Records Function.	9-19
9.3.6.3.1.1. Event File.	9-20
9.3.6.3.1.1.1. New Event Record.	9-20
9.3.6.3.1.1.2. New Event Station Record.	9-21
9.3.6.3.1.2. Station File.	9-21
9.3.6.3.1.3. Canister File.	9-21
9.3.6.3.1.4. Gauge File.	9-22
9.3.6.3.2. Update Records Function.	9-22
9.3.6.3.2.1. Event File.	9-23
9.3.6.3.2.2. Station File.	9-23
9.3.6.3.2.3. Canister File.	9-24
9.3.6.3.2.4. Gauge File.	9-24
9.3.6.3.3. Delete Records Function.	9-25
9.3.6.3.3.1. Event File.	9-25
9.3.6.3.3.2. Station File.	9-26
9.3.6.3.3.3. Canister File.	9-26
9.3.6.3.3.4. Gauge File.	9-27
9.3.6.3.4. Scan Records Function.	9-28
9.3.6.3.4.1. Event File.	9-28
9.3.6.3.4.2. Station File.	9-29
9.3.6.3.4.3. Canister File.	9-29
9.3.6.3.4.4. Gauge File.	9-30
9.3.6.4. Program Termination.	9-30

CHAPTER 10 "SCAN" USER MANUAL

10.1. GENERAL INFORMATION	10-2
10.1.1. Summary.	10-2
10.1.2. Environment.	10-2
10.1.3. References.	10-2
10.2. APPLICATION	10-3
10.2.1. Description.	10-3
10.2.1.1. When to Use this Program.	10-3
10.2.1.2. Program Functions.	10-3
10.2.2. Operation.	10-3
10.2.3. Equipment.	10-3

10.2.4. Structure.	10-3
10.2.5. Performance.	10-4
10.2.6. Inputs, Processing, and Outputs.	10-4
10.2.6.1. Inputs.	10-4
10.2.6.2. Processing.	10-4
10.2.6.2.1. Read the Parameter List.	10-4
10.2.6.2.2. Create the Scan List.	10-4
10.2.6.3. Outputs.	10-5
10.3. PROCEDURES AND REQUIREMENTS	10-6
10.3.1. Initiation.	10-6
10.3.1.1. The Procedure.	10-6
10.3.1.2. Getting a Copy of the Procedure.	10-6
10.3.1.3. Invoking the Procedure.	10-6
10.3.1.4. Compilation and Execution.	10-7
10.3.2. Input.	10-7
10.3.2.1. Requirements.	10-7
10.3.2.1.1. Origin.	10-7
10.3.2.1.2. Medium.	10-7
10.3.2.1.3. Restrictions.	10-7
10.3.2.2. Input Formats.	10-7
10.3.2.2.1. Length.	10-7
10.3.2.2.2. Format.	10-8
10.3.2.2.3. Labels.	10-8
10.3.2.2.4. Sequence.	10-8
10.3.2.2.5. Combination.	10-8
10.3.3. Sample Inputs.	10-8
10.3.3.1. The Input File.	10-8
10.3.3.1.1. The Identifier Block.	10-9
10.3.3.1.2. The Modification Records.	10-9
10.3.3.1.3. The Data Records.	10-9
10.3.3.2. The Procedure Call Statement.	10-9
10.3.4. Output.	10-9
10.3.4.1. Use.	10-9
10.3.4.2. Frequency.	10-10
10.3.4.3. Variations.	10-10
10.3.4.4. Destination/Disposition.	10-10
10.3.4.5. Medium.	10-10
10.3.4.6. Sample Outputs.	10-10
10.3.4.6.1. Definition.	10-10
10.3.4.6.2. Source.	10-10
10.3.4.7. At Program Termination.	10-12
10.3.5. Error and Recovery.	10-12

CHAPTER 11 "PLOT" USER MANUAL

11.1. GENERAL INFORMATION	11-2
11.1.1. Summary.	11-2
11.1.2. Environment.	11-2
11.1.3. References.	11-2
11.2. APPLICATION	11-3

11.2.1. Description.	11-3
11.2.1.1. When to Use this Program.	11-3
11.2.1.2. Program Functions.	11-3
11.2.2. Operation.	11-3
11.2.3. Equipment.	11-3
11.2.4. Structure.	11-4
11.2.5. Performance.	11-4
11.2.6. Inputs, Processing, and Outputs.	11-4
11.2.6.1. Inputs.	11-4
11.2.6.1.1. Description.	11-4
11.2.6.1.1.1. Input File Name.	11-4
11.2.6.1.1.2. Number of Frames for a Plot.	11-4
11.2.6.1.1.3. Smoothness of the Curve.	11-5
11.2.6.1.1.4. Start Time of the Plot.	11-5
11.2.6.1.1.5. Stop Time of the Plot.	11-5
11.2.6.1.1.6. Plot Device.	11-5
11.2.6.2. Processing.	11-5
11.2.6.2.1. Read the Parameter List.	11-5
11.2.6.2.2. Create the Plot File.	11-6
11.2.6.3. Outputs.	11-6
11.3. PROCEDURES AND REQUIREMENTS	11-7
11.3.1. Initiation.	11-7
11.3.1.1. The Procedure.	11-7
11.3.1.2. Getting a Copy of the Procedure.	11-7
11.3.1.3. Invoking the Procedure.	11-7
11.3.1.4. Compilation and Execution.	11-8
11.3.2. Input.	11-8
11.3.2.1. Procedure Parameters.	11-8
11.3.2.2. Requirements.	11-8
11.3.2.2.1. Frequency.	11-8
11.3.2.2.2. Origin.	11-8
11.3.2.2.3. Medium.	11-9
11.3.2.2.4. Restrictions.	11-9
11.3.2.2. Input Formats.	11-9
11.3.2.2.1. Length.	11-9
11.3.2.2.2. Format.	11-9
11.3.2.2.3. Labels.	11-9
11.3.2.2.4. Sequence.	11-9
11.3.3. Sample Inputs.	11-10
11.3.3.1. The Input File.	11-10
11.3.3.1.1. The Identifier Block.	11-10
11.3.3.1.2. The Modification Records.	11-10
11.3.3.1.3. The Data Records.	11-10
11.3.3.2. The Procedure Call Statement.	11-10
11.3.4. Output.	11-11
11.3.4.1. Use.	11-11
11.3.4.2. Variations.	11-11
11.3.4.3. Destination.	11-11
11.3.4.3.1. Line Printer.	11-11
11.3.4.3.2. Plotter.	11-11
11.3.4.3.3. Terminal.	11-11
11.3.4.4. Medium.	11-11
11.3.4.5. Disposition.	11-12

11.3.4.6 Sample Outputs.	11-12
11.3.4.7. At Program Termination.	11-16
11.3.5. Error and Recovery.	11-16

CHAPTER 12 "SPECTRA" USER MANUAL

12.1. GENERAL INFORMATION	12-2
12.1.1. Summary.	12-2
12.1.2. Environment.	12-2
12.1.3. References.	12-2
12.2. APPLICATION	12-3
12.2.1. Description.	12-3
12.2.1.1. When to Use this Program.	12-3
12.2.1.2. Program Functions.	12-3
12.2.2. Operation.	12-3
12.2.3. Equipment.	12-3
12.2.4. Structure.	12-4
12.2.5. Performance.	12-4
12.2.6. Inputs, Processing, and Outputs.	12-4
12.2.6.1. Inputs.	12-4
12.2.6.1.1. Description.	12-4
12.2.6.1.1.1. Input File Name.	12-4
12.2.6.1.1.2. The Start Time of the Noise Segment.	12-4
12.2.6.1.1.3. The End Time of the Noise Segment.	12-5
12.2.6.1.1.4. The Start Time of the Signal Segment.	12-5
12.2.6.1.1.5. The End Time of the Signal Segment.	12-5
12.2.6.1.1.6. The Block Size.	12-5
12.2.6.1.1.7. Plot Device.	12-5
12.2.6.2. Processing.	12-5
12.2.6.2.1. Read the Parameter List.	12-5
12.2.6.2.2. Calculate the Spectra.	12-6
12.2.6.2.2.1. Calculate the Periodograms.	12-6
12.2.6.2.3. The Spectra Plot.	12-6
12.2.6.3. Outputs.	12-6
12.3. PROCEDURES AND REQUIREMENTS	12-7
12.3.1. Initiation.	12-7
12.3.1.1. The Procedure.	12-7
12.3.1.2. Getting a Copy of the Procedure.	12-7
12.3.1.3. Invoking the Procedure.	12-7
12.3.1.4. Compilation and Execution.	12-8
12.3.2. Input.	12-8
12.3.2.1. Procedure Parameters.	12-8
12.3.2.2. Requirements.	12-8
12.3.2.2.1. Frequency.	12-8
12.3.2.2.2. Origin.	12-8
12.3.2.2.3. Medium.	12-8
12.3.2.2.4. Restrictions.	12-9
12.3.2.2. Input Formats.	12-9
12.3.2.2.1. Length.	12-9
12.3.2.2.2. Format.	12-9

12.3.2.2.3. Labels.	12-9
12.3.2.2.4. Sequence.	12-9
12.3.3. Sample Inputs.	12-9
12.3.3.1. The Input File.	12-9
12.3.3.1.1. The Identifier Block.	12-10
12.3.3.1.2. The Modification Records.	12-10
12.3.3.1.3. The Data Records.	12-10
12.3.3.2. The Procedure Call Statement.	12-10
12.3.4. Output.	12-10
12.3.4.1. Use.	12-10
12.3.4.2. Destination.	12-11
12.3.4.2.1. Line Printer.	12-11
12.3.4.2.2. Plotter.	12-11
12.3.4.2.3. Terminal.	12-11
12.3.4.3. Medium.	12-11
12.3.4.4. Disposition.	12-11
12.3.4.5 Sample Outputs.	12-11
12.3.4.6. Program Termination.	12-15
12.3.5. Error and Recovery.	12-15

CHAPTER 13

"FILTER" USER MANUAL

13.1. GENERAL INFORMATION	13-2
13.1.1. Summary.	13-2
13.1.2. Environment.	13-2
13.1.3. References.	13-2
13.2. APPLICATION	13-3
13.2.1. Description.	13-3
13.2.1.1. When to Use this Program.	13-3
13.2.1.2. Program Functions.	13-3
13.2.1.2.1. Parameter Specification.	13-3
13.2.1.2.2. Plot File.	13-3
13.2.1.2.3. Output File.	13-4
13.2.2. Operation.	13-4
13.2.3. Equipment.	13-4
13.2.4. Structure.	13-4
13.2.5. Performance.	13-5
13.2.6. Inputs, Processing, and Outputs.	13-5
13.2.6.1. Inputs.	13-5
13.2.6.1.1. Description.	13-5
13.2.6.1.1.1. Input File Name.	13-5
13.2.6.1.1.2. Output File Name.	13-5
13.2.6.1.1.3. Volume Serial Number of the Output Tape.	13-5
13.2.6.1.1.4. Highpass Filter Frequency.	13-6
13.2.6.1.1.5. Lowpass Filter Frequency.	13-6
13.2.6.1.1.6. Input Delta Time.	13-6
13.2.6.1.1.7. Output Delta Time.	13-6
13.2.6.1.1.8. Data Start Time.	13-6
13.2.6.1.1.9. Data End Time.	13-6
13.2.6.1.1.10. Data Arrival Time.	13-7
13.2.6.1.1.11. Plot Device.	13-7

13.2.6.2. Processing.	13-7
13.2.6.2.1. Read the Parameter List.	13-7
13.2.6.2.2. Preparation For Filtering the Data.	13-7
13.2.6.2.2.1. Building the Filters.	13-7
13.2.6.2.2.2. Getting the Data File.	13-7
13.2.6.2.3. Filtering the Acceleration Data.	13-8
13.2.6.2.4. Filtering the Velocity Data.	13-8
13.2.6.2.5. Filtering the Displacement Data.	13-8
13.2.6.3. Outputs.	13-8
13.3. PROCEDURES AND REQUIREMENTS	13-10
13.3.1. Initiation.	13-10
13.3.1.1. The Procedure.	13-10
13.3.1.2. Getting a Copy of the Procedure.	13-10
13.3.1.3. Invoking the Procedure.	13-10
13.3.1.4. Compilation and Execution.	13-11
13.3.2. Input.	13-11
13.3.2.1. Procedure Parameters.	13-11
13.3.2.2. Requirements.	13-11
13.3.2.2.2. Origin.	13-11
13.3.2.2.3. Medium.	13-12
13.3.2.2.4. Restrictions.	13-12
13.3.2.3. Input Formats.	13-12
13.3.2.3.1. Length.	13-12
13.3.2.3.2. Format.	13-12
13.3.2.3.3. Labels.	13-12
13.3.2.3.4. Sequence.	13-12
13.3.3. Sample Inputs.	13-13
13.3.3.1. The Input File.	13-13
13.3.3.1.1. The Identifier Block.	13-13
13.3.3.1.2. The Modification Records.	13-13
13.3.3.1.3. The Data Records.	13-13
13.3.3.2. The Procedure Call Statement.	13-13
13.3.4. Output.	13-14
13.3.4.1. Use.	13-14
13.3.4.2. Destination.	13-14
13.3.4.2.1. Line Printer.	13-14
13.3.4.2.2. Plotter.	13-14
13.3.4.2.3. Terminal.	13-14
13.3.4.3. Medium.	13-14
13.3.4.4. Disposition.	13-14
13.3.4.5 Sample Outputs.	13-15
13.3.4.5.1. Plot.	13-15
13.3.4.5.2. Output Data File.	13-15
13.3.4.6. At Program Termination.	13-26
13.3.5. Error and Recovery.	13-26

CHAPTER 14

"ROTATE" USER MANUAL

14.1. GENERAL INFORMATION	14-2
14.1.1. Summary.	14-2
14.1.2. Environment.	14-2

14.1.3. References.	14-2
14.2. APPLICATION	14-3
14.2.1. Description.	14-3
14.2.1.1. When to Use this Program.	14-3
14.2.1.2. Program Functions.	14-3
14.2.1.2.1. Plot File.	14-3
14.2.1.2.2. Output File.	14-3
14.2.2. Operation.	14-4
14.2.3. Equipment.	14-4
14.2.4. Structure.	14-4
14.2.5. Performance.	14-4
14.2.6. Inputs, Processing, and Outputs.	14-4
14.2.6.1. Inputs.	14-4
14.2.6.1.1. Description.	14-5
14.2.6.1.1.1. Input File Name.	14-5
14.2.6.1.1.2. Output File Name.	14-5
14.2.6.1.1.3. Volume Serial Number of the Output Tape.	14-5
14.2.6.1.1.4. The Event Number.	14-5
14.2.6.1.1.5. Plot Device.	14-5
14.2.6.2. Processing.	14-5
14.2.6.2.1. Read the Parameter List.	14-5
14.2.6.2.2. Initialize the Data Base.	14-6
14.2.6.2.3. Preparation For Rotating the Data.	14-6
14.2.6.2.3.1. Get The Data File.	14-6
14.2.6.2.3.2. Verify The Data File.	14-6
14.2.6.2.3.3. Read Data Base Records.	14-6
14.2.6.2.3.4. Read Station Orientations.	14-6
14.2.6.2.3.5. Write Data to the Output File.	14-7
14.2.6.2.3.6. Read All Files For The Station.	14-7
14.2.6.2.3.7. The Output Files.	14-7
14.2.6.2.4. Rotate the Data.	14-7
14.2.6.3. Outputs.	14-7
14.3. PROCEDURES AND REQUIREMENTS	14-8
14.3.1. Initiation.	14-8
14.3.1.1. The Procedure.	14-8
14.3.1.2. Getting a Copy of the Procedure.	14-8
14.3.1.3. Invoking the Procedure.	14-8
14.3.1.4. Compilation and Execution.	14-9
14.3.2. Input.	14-9
14.3.2.1. Procedure Parameters.	14-9
14.3.2.2. Requirements.	14-9
14.3.2.2.1. Frequency.	14-9
14.3.2.2.2. Origin.	14-9
14.3.2.2.3. Medium.	14-10
14.3.2.2.4. Restrictions.	14-10
14.3.2.3. Input Formats.	14-10
14.3.2.3.1. Length.	14-10
14.3.2.3.2. Format.	14-10
14.3.2.3.3. Labels.	14-10
14.3.2.3.4. Sequence.	14-10
14.3.3. Sample Inputs.	14-11
14.3.3.1. The Input File.	14-11

14.3.3.1.1. The Identifier Block.	14-11
14.3.3.1.2. The Modification Records.	14-11
14.3.3.1.3. The Data Records.	14-11
14.3.3.2. The Procedure Call Statement.	14-11
14.3.4. Output.	14-12
14.3.4.1. Use.	14-12
14.3.4.2. Destination.	14-12
14.3.4.2.1. Line Printer.	14-12
14.3.4.2.2. Plotter.	14-12
14.3.4.2.3. Terminal.	14-12
14.3.4.3. Medium.	14-12
14.3.4.4. Disposition.	14-13
14.3.4.5 Sample Outputs.	14-13
14.3.4.5.1. Plot.	14-13
14.3.4.5.2. Output Data File.	14-13
14.3.4.6. At Program Termination.	14-24
14.3.5. Error and Recovery.	14-24

CHAPTER 15

"PSRV" USER MANUAL

15.1. GENERAL INFORMATION	15-2
15.1.1. Summary.	15-2
15.1.2. Environment.	15-2
15.1.3. References.	15-2
15.2. APPLICATION	15-3
15.2.1. Description.	15-3
15.2.1.1. When to Use this Program.	15-3
15.2.1.2. Program Functions.	15-3
15.2.2. Operation.	15-3
15.2.3. Equipment.	15-3
15.2.4. Structure.	15-3
15.2.5. Performance.	15-4
15.2.6. Inputs, Processing, and Outputs.	15-4
15.2.6.1. Inputs.	15-4
15.2.6.1.1. Description.	15-4
15.2.6.1.1.1. Input File Name.	15-4
15.2.6.1.1.2. Start Time of the Data.	15-4
15.2.6.1.1.3. End Time of the Data.	15-4
15.2.6.1.1.4. The Damping Factor.	15-5
15.2.6.1.1.5. Number of Curves To Plot On Each Plot.	15-5
15.2.6.1.1.6. Number of Files To Process In The Input File.	15-5
15.2.6.1.1.7. The Minimum Frequency.	15-5
15.2.6.1.1.8. The Maximum Frequency.	15-5
15.2.6.2. Processing.	15-5
15.2.6.2.1. Initialize.	15-5
15.2.6.2.1.1. Read the Parameter List.	15-5
15.2.6.2.1.2. Verify The Parameter Values.	15-6
15.2.6.2.1.3. Determine Periods For Calculation.	15-6
15.2.6.2.2. The Input File.	15-6
15.2.6.2.2.1. Read the Data File.	15-6

15.2.6.2.2.2. Prepare For The Pseudo Relative Response Velocity Calculation.	15-6
15.2.6.2.2.3. Calculate the Pseudo Relative Response Velocity.	15-6
15.2.6.2.2.4. The Pseudo Relative Response Velocity Plot.	15-7
15.2.6.3. Outputs.	15-7
15.3. PROCEDURES AND REQUIREMENTS	15-8
15.3.1. Initiation.	15-8
15.3.1.1. The Procedure.	15-8
15.3.1.2. Getting a Copy of the Procedure.	15-8
15.3.1.3. Invoking the Procedure.	15-8
15.3.1.4. Compilation and Execution.	15-9
15.3.2. Input.	15-9
15.3.2.1. Procedure Parameters.	15-9
15.3.2.2. Requirements.	15-9
15.3.2.2.1. Frequency.	15-9
15.3.2.2.2. Origin.	15-9
15.3.2.2.3. Medium.	15-10
15.3.2.2.4. Restrictions.	15-10
15.3.2.3. Input Formats.	15-10
15.3.2.3.1. Length.	15-10
15.3.2.3.2. Format.	15-10
15.3.2.3.3. Labels.	15-10
15.3.2.3.4. Sequence.	15-10
15.3.3. Sample Inputs.	15-11
15.3.3.1. The Input File.	15-11
15.3.3.1.1. The Identifier Block.	15-11
15.3.3.1.2. The Modification Records.	15-11
15.3.3.1.3. The Data Records.	15-11
15.3.3.2. The Procedure Call Statement.	15-11
15.3.4. Output.	15-12
15.3.4.1. Use.	15-12
15.3.4.2. Variations.	15-12
15.3.4.3. Destination.	15-12
15.3.4.3.1. Line Printer.	15-12
15.3.4.3.2. Plotter.	15-12
15.3.4.3.3. Terminal.	15-12
15.3.4.4. Medium.	15-12
15.3.4.5. Disposition.	15-13
15.3.4.6. Sample Outputs.	15-13
15.3.4.6.1. Plot.	15-13
15.3.4.7. At Program Termination.	15-17
15.3.5. Error and Recovery.	15-17

CHAPTER 16

"VECTOR" USER MANUAL

16.1. GENERAL INFORMATION	16-2
16.1.1. Summary.	16-2
16.1.2. Environment.	16-2
16.1.3. References.	16-2
16.2. APPLICATION	16-3

16.2.1. Description.	16-3
16.2.1.1. When to Use this Program.	16-3
16.2.1.2. Program Functions.	16-3
16.2.1.2.1. Plot File.	16-3
16.2.1.2.2. Output File.	16-3
16.2.2. Operation.	16-3
16.2.3. Equipment.	16-3
16.2.4. Structure.	16-4
16.2.5. Performance.	16-4
16.2.6. Inputs, Processing, and Outputs.	16-4
16.2.6.1. Inputs.	16-4
16.2.6.1.1. Description.	16-4
16.2.6.1.1.1. Input File Name.	16-4
16.2.6.1.1.2. Output File Name.	16-4
16.2.6.1.1.3. Volume Serial Number of the Output Tape.	16-5
16.2.6.1.1.4. The Start Time of the Data.	16-5
16.2.6.1.1.5. The End Time of the Data.	16-5
16.2.6.1.1.6. PSRV Input Data.	16-5
16.2.6.1.1.7. Plot Device.	16-5
16.2.6.2. Processing.	16-6
16.2.6.2.1. Read the Parameter List.	16-6
16.2.6.2.2. Separate the Channel Files for a Station.	16-6
16.2.6.2.3. The Method For Doing the Vector Sum Calculation.	16-6
16.2.6.2.3.1. Determine the Data Segments For Vectorization.	16-6
16.2.6.2.4. Prepare For Vector Calculations.	16-7
16.2.6.2.4.1. Read the Data From The Scratch Files.	16-7
16.2.6.2.4.2. Extract The Data Segments.	16-7
16.2.6.2.5. Perform The Vector Sum Calculation.	16-7
16.2.6.2.6. Write to the Output File.	16-8
16.2.6.2.7. Write to the Plot File.	16-8
16.2.6.2.8. Calculate the Remaining Two Vector Sums.	16-8
16.2.6.2.9. Calculate the Three Vector Sums For Each Station In The Input Multifile.	16-8
16.2.6.3. Outputs.	16-8
16.3. PROCEDURES AND REQUIREMENTS	16-9
16.3.1. Initiation.	16-9
16.3.1.1. The Procedure.	16-9
16.3.1.2. Getting a Copy of the Procedure.	16-9
16.3.1.3. Invoking the Procedure.	16-9
16.3.1.4. Compilation and Execution.	16-10
16.3.2. Input.	16-10
16.3.2.1. Procedure Parameters.	16-10
16.3.2.2. Requirements.	16-10
16.3.2.2.1. Frequency.	16-10
16.3.2.2.2. Origin.	16-10
16.3.2.2.3. Medium.	16-11
16.3.2.2.4. Restrictions.	16-11
16.3.2.3. Input Formats.	16-11
16.3.2.3.1. Length.	16-11
16.3.2.3.2. Format.	16-11
16.3.2.3.3. Labels.	16-11
16.3.2.3.4. Sequence.	16-11
16.3.3. Sample Inputs.	16-12

16.3.3.1. The Input File.	16-12
16.3.3.1.1. The Identifier Block.	16-12
16.3.3.1.2. The Modification Records.	16-12
16.3.3.1.3. The Data Records.	16-12
16.3.3.2. The Procedure Call Statement.	16-12
16.3.4. Output.	16-13
16.3.4.1. Use.	16-13
16.3.4.2. Destination.	16-13
16.3.4.3.1. Line Printer.	16-13
16.3.4.2.2. Plotter.	16-13
16.3.4.2.3. Terminal.	16-13
16.3.4.3. Medium.	16-13
16.3.4.4. Disposition.	16-13
16.3.4.5 Sample Outputs.	16-14
16.3.4.5.1. Plot.	16-14
16.3.4.5.2. Output Data File.	16-14
16.3.4.6. At Program Termination.	16-19
16.3.5. Error and Recovery.	16-19

LIST of FIGURES

Figure 9.1	Summary Data Sheet.....	9-12
Figure 9.2	Recording Information.....	9-13
Figure 9.3	Track Sheet.....	9-14
Figure 9.4	Canister Log.....	9-15
Figure 9.5	Gauge Calibration.....	9-16
Figure 10.1	Scan Output.....	10-11
Figure 11.1	Vertical Plot.....	11-13
Figure 11.2	Radial Plot.....	11-14
Figure 11.3	Tangential Plot.....	11-15
Figure 12.1	Vertical Spectrum.....	12-12
Figure 12.2	Radial Spectrum	12-13
Figure 12.3	Tangential Spectrum.....	12-14
Figure 13.1	Vertical Acceleration.....	13-16
Figure 13.2	Vertical Velocity.....	13-17
Figure 13.3	Vertical Displacement.....	13-18
Figure 13.4	Radial Acceleration.....	13-19
Figure 13.5	Radial Velocity.....	13-20
Figure 13.6	Radial Displacement.....	13-21
Figure 13.7	Tangential Acceleration.....	13-22
Figure 13.8	Tangential Velocity.....	13-23
Figure 13.9	Tangential Displacement.....	13-24
Figure 13.10	Output File Scan.....	13-25
Figure 14.1	Vertical Acceleration.....	14-14
Figure 14.2	Vertical Velocity.....	14-15
Figure 14.3	Vertical Displacement.....	14-16
Figure 14.4	Rotated Radial Acceleration.....	14-17
Figure 14.5	Rotated Radial Velocity.....	14-18
Figure 14.6	Rotated Radial Displacement.....	14-19
Figure 14.7	Rotated Tangential Acceleration.....	14-20
Figure 14.8	Rotated Tangential Velocity.....	14-21
Figure 14.9	Rotated Tangential Displacement.....	14-22
Figure 14.10	Output File Scan.....	14-23
Figure 15.1	Vertical PSRV.....	15-14
Figure 15.2	Radial PSRV.....	15-15
Figure 15.3	Tangential PSRV.....	15-16
Figure 16.1	Acceleration Vector Sum.....	16-15
Figure 16.2	Velocity Vector Sum.....	16-16
Figure 16.3	Displacement Vector Sum.....	16-17
Figure 16.4	Output File Scan.....	16-18

"ACCESS"
USER MANUAL

CHAPTER 9
"ACCESS"
USER MANUAL

"ACCESS"
USER MANUAL

9.1. GENERAL INFORMATION

9.1.1. Summary.

The "ACCESS" program is designed to allow access to the WTSI data base to perform any or all of four functions:

- a) To allow new WTSI data to be entered into the data base, as is the case when adding a new set of records.
- b) To allow existing data base records to be modified or updated to contain most current information.
- c) To allow existing data base records to be deleted as is the case when erroneous data has been entered into the data base.
- d) To allow existing data base records to be inspected, or scanned.

9.1.2. Environment.

This project is sponsored by the NNWSI project. Its development is the responsibility of the Ground Motion and Seismic Division (7111) of the Field Sciences Department (7110). Members of the project within Division 7111 are the sole users of the "ACCESS" program.

The "ACCESS" program is implemented on Sandia National Laboratories' CDC CYBER 180-855 model computer, located in the Central Computing Facility in Sandia's primary technical area, Area I, in building 800. The CYBER 180-855 operates in a dual state, which allows two operating systems to run concurrently. The "ACCESS" program runs in the NOS environment. The current operating system version is 2.4.2 level 642.

9.1.3. References.

- 9.1.3.1. National Bureau of Standards, "Guidelines for Documentation of Computer Programs and Automated Data Systems," Federal Information Processing Standards (FIPS) Publication #38, February, 1976.

"ACCESS" USER MANUAL

9.2. APPLICATION

9.2.1. Description.

The "ACCESS" program provides interactive access to the WTSI data base files. The data base consists of four files that contain information regarding the events, stations, canisters, and gauges that pertain to the NNWSI project. The interactive methods employed by the program provide easy access to any of the four data base files to perform any of several functions.

9.2.1.1. When to Use this Program.

The technical personnel on-site at the Nevada Test Site (NTS) transmit the event, station, canister, and gauge information to Division 7111 as it becomes available. The "ACCESS" program is used as often as necessary to maintain the most current information in the data base files and can be referred to at any time WTSI information is required.

9.2.1.2. Program Functions.

This interactive program provides access to the WTSI data base for adding, updating, deleting, and scanning data base records. These data base records are presented in the form of screen panels on the terminal. The program accesses the panels through the NOS screen PDU, which interfaces to the auxiliary keypad as well as the primary keyboard.

9.2.2. Operation.

Only those members of the Ground Motion and Seismic Division (7111) who are working on the project and have need to access the data base are authorized to run the "ACCESS" program. At the present time, there are two such users.

9.2.3. Equipment.

The "ACCESS" program resides on the NOS side of the dual state CDC CYBER 180-855. The CYBER 180-855, in the NOS environment, is configured with 1.4 million words (11.2 million bytes) of memory, and eight 885 disks, each with 2 spindles, for a total of 5 giga-bytes of online disk space.

9.2.4. Structure.

All software for the "ACCESS" program is written in FORTRAN 5 and executed under the control of CDC's Network Operating System. The screen panels that the program uses were designed as lines of text. The PDU was used to create a compiled version of each panel; these compiled

versions are stored in a library and are accessible by the program as it executes.

The "ACCESS" program is written in a modular fashion. Each routine performs one basic function or set of related functions, thereby providing a program that is easy to read and modify. The main program interprets the user's selection and directs the appropriate routine to verify and process the input data. When the user's input has been dealt with appropriately, the main program determines the next function to be performed and initiates the proper routine for processing.

9.2.5. Performance.

Currently, the data base access program exists and is run on the NOS side of the dual state CYBER 180-855 computer. Eventually, this program will be migrated to run on the other CYBER 180-855 operating system, NOS/VE. There exists no method of successfully migrating the program to the NOS/VE at the present time. The screen panels which the program accesses during execution cannot be executed on NOS/VE, as the operating system does not yet include the capability to design screen panels.

9.2.6. Data Base.

The "ACCESS" program requires the four WTSI data base files be available during program execution.

9.2.6.1. The Event File.

Of the four data base files, the event file is the only direct access file. It describes all events pertinent to the NNWSI project in a chronological, detailed format and is updated as necessary to add new data as it becomes available. Its permanent file name is "EVENT."

9.2.6.2. The Station File.

The station file is an indirect access file. It describes all stations pertinent to the NNWSI project in a chronological, detailed format and is updated as necessary to add new data as it becomes available. Its permanent file name is "STATION."

9.2.6.3. The Canister File.

The canister file is an indirect access file. It describes all canisters pertinent to the NNWSI project in a chronological, detailed format and is updated as necessary to add new data as it becomes available. Its permanent file name is "CANSTR."

9.2.6.4. The Gauge File.

The gauge file is an indirect access file. It describes all gauges pertinent to the NNWSI project in a chronological, detailed format and is updated as necessary to add new data as it becomes available. Its permanent file name is "GAUGE."

9.2.7. Inputs, Processing, and Outputs.

9.2.7.1. Inputs.

The "ACCESS" program requires as input only four data files, known also as the data base files, as they comprise the WTSI data base. The screen panel library contains the screen panel definitions in their compiled state. Given that the "ACCESS" program is interactive, the processing is best described by including references to the program outputs - the screen panels.

9.2.7.2. Interacting with the Program.

When a screen panel is displayed, the user is expected to enter a response that will direct the program to perform either a function or an operation. The following describes how to transmit the desired responses to the program.

Some screens display a list of functions as well as a list of operations that may be performed. The desired function is selected using the number keys "1" through "9" on the keyboard. The desired operation is selected using the "1" through "9" keys on the auxiliary keypad on the keyboard, although the screen panels refer to these keys as "F1", "F2", etc. It is important to note that some keypads may have keys that are marked "F1", "F2", through "F9"; these keys are not always the appropriate keys. It may be necessary to consult the appropriate terminal user's manual to determine which keys are the "1" through "9" keys on the keypad. These are the only keys that should be used to enter the operations.

As the user selects the various functions and operations, records from the data base files are displayed and may be altered or deleted, and new records can be added to a data base file. By repeatedly selecting from the functions and operations, the user can access every record in each of the data base files.

9.2.7.3. Outputs.

If any changes are made to the "STATION", "CANSTR", or "GAUGE" data base files (ref.: section 9.2.6.), certain steps must be taken upon program completion to preserve any changes made. Replacing the permanent copy of these files with the updated local copy will make

"ACCESS"
USER MANUAL

permanent any changes made to the file. The "EVENT" file retains any changes made, as it, rather than a copy, is accessed during the program.

9.3. PROCEDURES AND REQUIREMENTS

9.3.1. Initiation.

9.3.1.1. Locating the Files Into the User File Space.

The program source, the data base files, and the screen panel definition library are resident in disk space allocated to a particular member of the Ground Motion and Seismic Division (7111). They must be retrieved from that permanent area and copied into the local file space area of the user desiring to execute the program. The following describes the steps necessary to copy the files into local file space for loading the program:

```
GET,STATION/UN=MLSANDE  
GET,CANSTR/UN=MLSANDE  
GET,GAUGE/UN=MLSANDE  
ATTACH,EVENT/UN=MLSANDE  
GET,ACCESS/UN=MLSANDE  
GET,PANELIB/UN=MLSANDE
```

These commands will work properly only if each of the files has been declared "public", or available for access by a user other than the owner. This is the current state of each of the files as they exist in the "MLSANDE" disk space.

9.3.1.2. Compiling The Program.

The source program is compiled using the following command:

```
FTN5,I=ACCESS,L=ACCLIST,E=ACCERR,B=ACCBIN
```

9.3.1.3. Loading The Program Into Memory.

After successful compilation of the program source, the program can be loaded into memory. This load sequence assumes that the binary of the program source is in a file named "ACCBIN." The two system libraries are specified during the load sequence. The following describes the steps necessary to load the program for execution:

```
LOAD,ACCBIN  
LDSET,LIB=SFLIB/SRVLIB  
LDSET,MAP=LOADMAP  
NOGO,ACCLGO
```


9.3.1.4. Executing The Program.

To begin program execution, enter:

ACCLGO

As the program executes, the user is expected to respond to the series of panels that appear on the screen. A small area on the screen in each panel is reserved to indicate the control keys and their functions. When the user desires to enter data into the data base, the screen is accompanied by messages indicating the format of the data to be entered as well as, in some cases, supplying a few examples. This method makes the program essentially self-explanatory.

9.3.2. Input.

9.3.2.1. Requirements.

9.3.2.1.1. Frequency.

The primary reason for running this program is to enter data for a new event, but this program should be run as often as necessary to maintain the most current information in the data base files. Other situations that necessitate execution of the "ACCESS" program include, but are not limited to, adding a new station, calibrating a gauge, removing a canister, and removing a gauge.

9.3.2.1.2. Origin.

The technical personnel on-site at the NTS transmit the event, station, canister, and gauge information to Division 7111 as it becomes available.

9.3.2.1.3. Medium.

This interactive program receives and displays data through the use of screen panels on the terminal. All interactions with the program are accomplished using the keys on the primary keyboard and the auxiliary keypad, which direct and control the flow of the program.

9.3.2.1.4. Restrictions.

There are no security restrictions placed on the operation of the "ACCESS" program, as it resides in the open partition of the CYBER 180-855 and all data is considered unclassified. Execution of the program and access to the data base files is limited only by the lack of knowledge concerning the user's catalog in which the files reside.

9.3.2.2. Input Formats.

The "ACCESS" program is an interactive program that accepts and displays data base information on the terminal screen in the form of a series of panels. Different panels are displayed depending on the data to be entered or accessed. The screen panels are designed to accept only valid data values for the parameters in the records of the data base files, and the program will display messages regarding incorrect entries.

9.3.2.2.1. Length.

The number of underline characters appearing on the screen next to the name of the parameter value determines the length of a parameter value. The parameter value does not, however, have to completely fill the allotted space.

9.3.2.2.2. Format.

The parameter value may be entered anywhere in the space provided, and does not have to be left-justified, unless the number of characters in the parameter value is exactly equal to the number of spaces provided.

9.3.2.2.3. Labels.

The parameter names that appear on the screen are exactly the same names that appear on the data input forms, described in section 9.3.3.

9.3.2.2.4. Sequence.

The parameter values may be entered on the screen panel in any order desired.

9.3.2.2.5. Combination.

Any combination of upper and lower case characters may be entered, as the program converts all characters to upper case. Some parameter values require a combination of characters and numbers; when the correct combination is not entered, the program displays a message indicating the correct format of the parameter value. Some parameter values consist entirely of numbers. The values that are quantitative, such as the elevation, depth, coordinates, and amplification must be entered as real numbers, that is, including a decimal point.

9.3.3. Sample Inputs.

Different forms are used to indicate the different types of data that should be entered into the data base. The following describes the layout of the forms that contain data to be entered into the data base.

9.3.3.1. Track Sheets.

Track sheets are received from the personnel in the field after an event has occurred. Track sheets contain all the information regarding an event with respect to the event name, date, coordinates, location, and to the recording stations, tape recorders used, tracks and frequencies of the stations' data recorded on tape, canisters and gauges in place at each station, and coordinates and location of each station. This information is taken directly from the track sheets and entered into the appropriate data base file.

Figure 9.1 is an example of the summary data sheet included in the set of track sheets. It contains information on the recorders used to record the data, such as the name and date of the event, the recorder serial numbers, the location of each of the recorders used, the start and stop times of the tape and the data, the time of the 10-second calibration. Figure 9.2 shows the serial numbers of the recorder machines used, the location of the recorders, and the tracks on the tapes on which the stations' data reside. There is one track sheet of information for each station for which data was recorded for the event. Figure 9.3 is an example of a station's track sheet. It contains all the information for a station regarding the particular event, e.g., the location of the station, the canister and gauges installed at the station, the calibration of each gauge, the track on the tape that the station's data is recorded, and the frequencies on which the data from each gauge is recorded.

9.3.3.2. Canister Logs.

Canister logs are received from the personnel in the field whenever a change is made to a canister. A canister sheet contains information regarding a specific canister. This can be regarding the station where the canister is installed, the canister itself, or the gauges in the canister.

Figure 9.4 is an example of a canister log sheet. The items that are to be entered in the data base are indicated by an asterisk.

9.3.3.3. Accelerometer Calibration Reports.

Gauge calibration reports are received from the personnel in the field whenever a gauge is recalibrated or a new gauge is calibrated. A calibration report sheet contains information regarding a specific

"ACCESS"
USER MANUAL

gauge, including the serial number, date of calibration, and outputs in volts for various inputs.

Figure 9.5 is an example of a gauge calibration report sheet. The items that are to be entered in the data base are indicated by an asterisk.

"ACCESS"
USER MANUAL

Figure 9.1

PAGE ____ OF ____

SUMMARY SHEET

TEST NO.: LAB 100 TEST NAME: _____ TEST DATE: 9-30-86

RECORDING SITE: NR03 LOCATION: BLD 3153 RECORDING DATE: 9-30-86

TAPE HEAD CONFIGURATION: (IRIG) TOTAL NUMBER OF TAPES: 3

CALIBRATION: STEP CAL MODE: DC CAL DURATION: 10 SEC STEPS: GAGE, 0v, 2.5v

DATA START TIME: 15:30:00 LENGTH OF RECORDING: EOT

TIME OF RANGE CHANGE: _____ HRS. _____

					MACH. NO. 1		MACH. NO. 2		MACH. NO. 3				
					TRANSPORT NO.		TRANSPORT NO.		TRANSPORT NO.				
					<u>8210707</u>		<u>8180693</u>		<u>1289</u>				
					TRANSPORT TYPE		TRANSPORT TYPE		DATA TAPE				
					<u>Amper</u>		<u>Amper</u>		<u>TAPE</u>		REWOUND		
											NOTES		
TAPE NO.	TAPE I.P.S.	DATE	REF FREQ	OPER. INITIAL	START	STOP	START	STOP	ORIG.	DUB.	YES	NO	
1	60	9-30-86	100 KHz	<u>QZ</u>	15:27:00	15:40:30			<u>✓</u>		<u>✓</u>		ON OFF POST CAL: 15:25:00 CAL: 15:35:10
2	2	9-30-86	100 KHz	<u>QZ</u>			15:27:00	15:40:30	<u>✓</u>		<u>✓</u>		POST CAL: 15:35:00 CAL: 15:35:10
3	100	9-30-86	—	<u>QZ</u>			15:27:00	15:40:30	<u>✓</u>		<u>✓</u>		
4													
5													
6													
7													
8													
9													
10													
11													
12													
13													
14													
15													
16													

Figure 9.2

LABQUARK

EVENT

Date: 9-30-86

RECORDING LOCATION NADA - BLD 3153

Tape Start Time 15:27:00
Cal On 15:35:00 Cal Off 15:35:10
Data Start Time 15:30:00
Tape Stop Time 15:40:30

Tape Machine 1289

Track	Station No.	Location
1	STA J	
2	WI-29	
3	STA K	
4	STA E	
5	STA I	
6	STA G	
7	WI-10	
8	STA F	
9	WI-26	
10	WI-30	
11	WI-7	
12	STA A	
13	WI-25	
14	WI-28	

Track	Station No.	Location
15	WI-6	
16	STA H	
17		
18		
19		
20		
21		
22		
23		
24		
25		
26		
27		
28	TIMING.	

Comments: WI-10 CAME ON NORMALLY AND LOOKED NORMAL
AT -5 MINUTES, AT APPROXIMATELY -3 MINUTES DATA WAS LOST
FROM WI-10. POST SHOT CHECK ON WI-10 INDICATED AN INTERMITTENT
RECEIVER WAS THE CAUSE FOR DATA LOSS ON WI-10.
STATION "K" THE AV GAGE APPEARS TO HAVE STARTED OSCILLATING.

"ACCESS"
USER MANUAL

Figure 9.3

TRACK SHEET

PAGE: ____ OF ____

TEST NO: .		TEST NAME: LABOUR		TEST DATE: 9-30-86														
STATION INFORMATION		CANISTER INFORMATION		T.M. BOX INFORMATION														
STATION NO. WI-6 ETS1		ACCEL. CAN. NO. 23		FREQ. 2250.5														
STATION LOCATION NRDS		NO. 25																
STA N COORDINANTS: N. 758,081		E. 604,499		ELEV. 3806.2														
CAL START TIME:		CAL TYPE:		STEP CAL: 0v, 2.5v														
DIST FROM G2:		CAL DURATION: 10 SEC		CAL MODE: DC REVISED 12-10-85														
CANISTER ORIENTATION; RADIAL TOWARD: TRUE EAST				RANGE CHANGE TIME DAY ____ TIME ____														
RECORD INFORMATION		EVENT TIME GAGE DATA																
TRACK NO.	CHAN. ID	FREQ.	DEV	MFGR GAGE TYPE	SER NO	GAGE V/G	G RANGE	AMP V/G	AMP NO	AMP GAIN	SYSTEM V/G	V/CM	GAGE V/G	G RANGE	AMP V/G	AMP NO	AMP GAIN	SYSTEM V/G
15	AV	70	7.5%	303T102	3593	4.963	.5037	0 0	1	4.963					0 0			
	270°	52.5		303 B	5721	4.908	.5094	0 0	1	4.908					0 0			
	0°	40		303B	5098	5.024	.4976	0 0	1	5.024					0 0			
	AV	30		303T102	3593	4.963	.5004	0 0	10	49.63					0 0			
	270°	22		303B	5721	4.908	.5009	0 0	10	49.08					0 0			
	0°	14.5		303B	5098	5.024	.0498	0 0	10	50.24					0 0			
		10.5		303T102	3593	4.963	.0098	0 0	51.1	253.6					0 0			
	70°	7.35		303B	5721	4.908	.0099		51.1	250.8								
	0°	5.4		303B	5098	5.024	.0097		51.1	256.7								
13	REF			100 KHz														
14	IRIG B	22.0		ORIGINATES NRDS FIRE ST.	REMARKS: INSTALLED RECALIBRATED GAGES 10/30/84 INSTALLED RECALIBRATED GAGES 12-6-85 30,22,14.5 CHANNELS HAVE INVERTED OUTPUTS													
14	IRIG B	30.0		ORIGINATES NRDS FIRE ST.														
14	VOICE	40.0		MTS NET 8														
14	ETG B	52.5	7.5%	ORIGINATES NRDS FIRE ST.														
14	IRIG B	70		ORIGINATES NRDS FIRE ST.														

OPERATOR: J. Lee	DATE: 9-30-86

Figure 9.4

CANISTER LOG

5TA-A

Canister # 43 Date Checked 7/10/85 Employee RKZ

	SERIAL #	RL	RG	g Range	V per g	+V	-V	-2 gV
AVHI g (Range Change)	<u>356</u>	<u>3927.2</u>	<u>~</u>	<u>50165</u>	<u>4.9835</u>	<u>4.995</u>	<u>4.972</u>	
AVLO g								
AR	<u>459</u>	<u>3704.6</u>	<u>~</u>	<u>50065</u>	<u>4.9935</u>	<u>4.975</u>	<u>5.012</u>	
AT	<u>471</u>	<u>3921.1</u>	<u>~</u>	<u>49840</u>	<u>5.0160</u>	<u>5.031</u>	<u>5.001</u>	

List Canister assignments and any changes. Initial and date each entry.

7/10/85 All gages are type RA1400-AA01-0. See
incoming inspection and field calibration
reports on these gages for more information.
Canister bench checked and calibrated
using TM box #18. RKZ
7/10/85 This canister replaces canister #20. Log
dated 4/13/84. RKZ
12/9/85 Calibration check was made using TM
box #18. All readings fell within ± 5 mV of
the initial ones above. RKZ

Figure 9.5

Canister #40 AV
STA-1600
9/4/85

File: 013000.FRM
E.G.S. G. DEPT. 1262 SERVO ACCELEROMETER FIELD CALIBRATION REPORT

MANUFACTURER: PART NO.: SERIAL NO.: USER: PROGRAM: LOCATION:
SUNDRAND 979-1300-007 209 J.L.R.
GAGE RANGE: MODEL NO.: IS ALL TEST EQUIPMENT TO BE USED IN-DATE? (See Test Equipment List.) YES NO
Configuration: RANGE, 1g; FM116-6 Filter OFF.
Calibration performed using Endevco SC116-2M 'g' Monitor.

Local gravity: 979.58671
Gravity correction factor: 1.00115
FREQ. RESPONSE @ 1 g RMS
Level: SC116-2M Output Freq. (Hz) (Volts (RMS)) (V/g (peak))

0	-0.0540 V	15	4.981	5.004	.
+0.1	0.5308 V	35	5.031	5.109	.
-0.1	0.5393 V	50	5.135	4.896	.
+0.2	0.9848 V	100	4.974	4.990	.
-0.2	1.0287 V	200	4.974	4.974	.
+0.5	2.4769 V	300	4.974	4.974	.
-0.5	2.5340 V	400	4.974	4.974	.
+1	4.974 V	500	4.974	4.974	.
-1	5.0504 V				

Noise Test: 0.100 Volts (Broadband) 90-degree Phase Shift
Volts (50 Hz filter) measured at 1190 Hz
Current Scale Factor (ISF) = 4.9812 mA/g
Load Resistance (RL) = 15.8 ohms
Calc. Sensitivity (SC) = 5.0331 V/g (peak) [SC = ISF x RL/1000]

Technician: Date: 13 Nov 84
Checked by: Date: 13 Nov 84
29 NOV 82

"ACCESS"
USER MANUAL

9.3.4. Output.

9.3.4.1. Use.

The data base files are the only output from the "ACCESS" program. They are referenced only by the "ACCESS" program and the data base back-up program that is run frequently to ensure that current copies of the files are available.

9.3.4.2. Frequency.

The "ACCESS" program is executed as often as is necessary to maintain the most current information in the data base files.

The back-up program is run each time the data base is updated. It reads the data base files and generates a dump of each file. Once a year, the back-up program is run and the dumps of the data base files are sent to Barry M. Schwartz, Division 6313, Sandia National Laboratories.

9.3.4.3. Medium.

The data base files are maintained and stored in permanent disk file space as described in section 9.3.1.1. The data base dump files are also maintained in this disk space. Hard-copy listings of the data base dump files are mailed to Barry M. Schwartz, as described in section 9.3.4.2.

9.3.4.4. Disposition.

Upon completion of the "ACCESS" program, the updated data base files should be returned to the permanent file space in the owner's catalog.

Several steps must be taken upon program completion to preserve any changes made to the data base. The following describes the steps necessary to replace the "CANSTR", "STATION", and "GAUGE" files with their updated local versions:

```
REPLACE,STATION=STATION/UN=MLSANDE  
REPLACE,CANSTR=CANSTR/UN=MLSANDE  
REPLACE,GAUGE=GAUGE/UN=MLSANDE
```

Only those files that were modified need to be replaced with their local file space versions. If no changes are made, or if the changes made to the "CANSTR", "STATION", and "GAUGE" files are not to be made permanent, the file replacement process can be omitted.

The "EVENT" file retains any changes made, as it, rather than a copy, is accessed during the program.

When program execution is completed, it is also necessary to remove the local copies of the program source, the program binary, the "EVENT" data base file, and the screen panel library from the user's file space. This is accomplished using the following command:

RETURN,ACCESS,ACCBIN,EVENT,PANELIB

9.3.4.5. Output.

The data base files are the only output from the "ACCESS" program, and they are stored in binary format. They are not accessed again until the "ACCESS" program is executed or the backup program is run.

9.3.5. Error and Recovery.

There are no special restart procedures. The program will run an unlimited number of times. In the case of an error in data entry, the program generates messages that are displayed on the screen, indicating the correct format for the particular parameter value.

9.3.6. File Query.

The "ACCESS" program is an interactive data entry program. The method of entering the data into the screen panels that appear on the screen is described in detail in section 9.2.7.2. The steps necessary to initiate the program are described in section 9.3.1. When a screen panel is displayed, the user is expected to enter a response that will direct the program to perform either a function or an operation. The method of entering data is also described in section 9.3.1.

9.3.6.1. Selecting a Data Base File.

When program execution begins, a screen panel is displayed directing the user to select a data base file to access. The user may choose from the following files:

1. Event
2. Station
3. Canstr
4. Gauge

and from the following operations:

"ACCESS" USER MANUAL

F1. enter
F7. help
F9. quit

The "enter" operation transmits the selected file to the program. The "help" operation requests that a message offering some assistance be displayed on the screen. The "quit" operation terminates program execution.

9.3.6.2. Selecting A Data Base Function.

After a data base file is selected for access, a screen panel is displayed directing the user to choose a function to be performed on that file. The following functions are displayed:

1. Add Records
2. Update Records
3. Delete Records
4. Scan Records
5. Exit

and the following operations are displayed:

F1. enter
F7. help
F9. quit

The "enter", "help", and "quit" operations are described in section 9.3.6.1.

9.3.6.3. Data Base Functions.

9.3.6.3.1. Add Records Function.

When this function is selected, the screen panel displayed depends on the data base file selected, as either an event number, a station name, a canister name, or a gauge name is requested and is used as a key to locate the appropriate data base records. In each case, the operations are the same.

F1. enter
F5. function
F7. file
F9. quit

The "function" operation allows the user to change data base functions, and the screen described in section 9.3.6.2 will be displayed. The "file" operation allows the user to change data base files, and the screen described in section 9.3.6.1 will be displayed. The "enter" and "quit" operations are described in section 9.3.6.1.

9.3.6.3.1.1. Event File.

The user must specify if a new event is to be added to the data base, or if a new event station is to be added. The panel displayed makes the entry of this information simple and straightforward, asking for an "E" or an "S" to be entered. The operations to choose from are as follows:

- F1. enter
- F5. match
- F7. help
- F9. quit

The "enter", "help", and "quit" operations are described in section 9.3.6.1. The "match" operation, like the "help" operation, requests that messages offering assistance be displayed on the screen.

9.3.6.3.1.1.1. New Event Record.

When a new event record is to be added to the data base file, a screen panel permitting entry of event data is displayed. The operations to choose from are:

- F1. next event
- F2. next station
- F3. function
- F4. file
- F5. match
- F7. help
- F9. quit

The "next event" operation allows the user to add another event to the data base, and a new screen will appear for data entry. The "next station" operation allows the user to add a new event station to the data base, and a screen will appear for data entry. The "function" and "file" operations are described in section 9.3.6.3.1. The "match", "help", and "quit" functions are described in section 9.3.6.3.1.1.

9.3.6.3.1.1.2. New Event Station Record.

When a new event station record is to be added to the data base file, a screen panel permitting entry of event station data is displayed. The operations to choose from are:

- F1. next station
- F2. next event
- F3. function
- F4. file
- F5. match
- F7. help
- F9. quit

The "next station" operation allows the user to add a new event station to the data base, and a screen will appear for data entry. The "next event" operation allows the user to add another event to the data base, and a new screen will appear for data entry. The "function" and "file" operations are described in section 9.3.6.3.1; the "match", "help", and "quit" operations are described in section 9.3.6.3.1.1.

9.3.6.3.1.2. Station File.

When a new station record is to be added to the data base file, a screen panel permitting entry of the station data is displayed. The operations to choose from are:

- F1. next station
- F3. function
- F4. file
- F5. match
- F7. help
- F9. quit

The "next station" operation allows the user to add a new station to the data base, and a screen will appear for data entry. The "function" and "file" operations are described in section 9.3.6.3.1; the "match", "help", and "quit" operations are described in section 9.3.6.3.1.1.

9.3.6.3.1.3. Canister File.

When a new canister record is to be added to the data base file, a screen panel permitting entry of the canister data is displayed. The operations to choose from are:

"ACCESS"
USER MANUAL

- F1. next canister
- F3. function
- F4. file
- F5. match
- F7. help
- F9. quit

The "next canister" operation allows the user to add a new canister to the data base, and a screen will appear for data entry. The "function", and "file" operations are described in section 9.3.6.3.1; the "match", "help", and "quit" functions are described in section 9.3.6.3.1.1.

9.3.6.3.1.4. Gauge File.

When a new gauge record is to be added to the data base file, a screen panel permitting entry of the gauge data is displayed. The operations to choose from are:

- F1. next gauge
- F3. function
- F4. file
- F5. match
- F7. help
- F9. quit

The "next gauge" operation allows the user to add a new gauge to the data base, and a screen will appear for data entry. The "function" and "file" operations are described in section 9.3.6.3.1; the "match", "help", and "quit" functions are described in section 9.3.6.3.1.1.

9.3.6.3.2. Update Records Function.

When this function is selected, the screen panel displayed depends on the data base file selected, as either an event number, a station name, a canister name, or a gauge name is requested and is used as a key to locate the appropriate data base records. In each case, the operations are the same.

- F1. enter
- F5. function
- F7. file
- F9. quit

The "function" and "file" operations are described in section 9.3.6.3.1; the "enter" and "quit" operations are described in section 9.3.6.1.

9.3.6.3.2.1. Event File.

A panel displaying an error message appears if the event is not located, and the user is allowed to enter another event number. When the records for the event to be accessed are located, the record containing data for the event is displayed. The user is allowed to make changes to the record displayed on the screen. The operations to choose from are as follows:

- F1. next in chain
- F2. key change
- F3. function
- F4. file
- F5. match
- F7. help
- F9. quit

The "next in chain" operation allows the user to view the record of the station associated with the event. The record will appear on the screen for inspection and updating. By repeating this operation, the user can view every station record associated with the specified event. The "key change" operation allows the user to return to the panel described above to enter another event number. The "function" and "file" operations are described in section 9.3.6.3.1; the "match", "help", and "quit" functions are described in section 9.3.6.3.1.1.

9.3.6.3.2.2. Station File.

A panel displaying an error message appears if the station is not located, and the user is allowed to enter another station number. When the records for the station to be accessed are located, a record containing data for the station is displayed. The user is allowed to make changes to the record displayed on the screen. The operations to choose from are as follows:

- F1. next in chain
- F2. key change
- F3. function
- F4. file
- F5. match
- F7. help
- F9. quit

The "next in chain" operation allows the user to view another record of data for the specified station. The record will appear on the screen for inspection and updating. By repeating this operation, the user can view every station record associated with the specified station. The "key change" operation allows the user to return to the panel described above to enter another station name. The "function" and "file" operations are described in section 9.3.6.3.1; the "match", "help", and "quit" functions are described in section 9.3.6.3.1.1.

9.3.6.3.2.3. Canister File.

A panel displaying an error message appears if the canister is not located, and the user is allowed to enter another canister number. When the records for the canister to be accessed are located, a record containing data for the canister is displayed. The user is allowed to make changes to the record displayed on the screen. The operations to choose from are as follows:

- F1. next in chain
- F2. key change
- F3. function
- F4. file
- F5. match
- F7. help
- F9. quit

The "next in chain" operation allows the user to view another record of data for the specified canister. The record will appear on the screen for inspection and updating. By repeating this operation, the user can view every canister record associated with the specified canister. The "key change" operation allows the user to return to the panel described above to enter another canister name. The "function" and "file" operations are described in section 9.3.6.3.1; the "match", "help", and "quit" functions are described in section 9.3.6.3.1.1.

9.3.6.3.2.4. Gauge File.

A panel displaying an error message appears if the gauge is not located, and the user is allowed to enter another gauge number. When the records for the canister to be accessed are located, a record containing data for the gauge is displayed. The user is allowed to make changes to the record displayed on the screen. The operations to choose from are as follows:

- F1. next in chain
- F2. key change
- F3. function

- F4. file
- F5. match
- F7. help
- F9. quit

The "next in chain" operation allows the user to view another record of data for the specified gauge. The record will appear on the screen for inspection and updating. By repeating this operation, the user can view every gauge record associated with the specified gauge. The "key change" operation allows the user to return to the panel described above to enter another gauge name. The "function" and "file" operations are described in section 9.3.6.3.1; the "match", "help", and "quit" functions are described in section 9.3.6.3.1.1.

9.3.6.3.3. Delete Records Function.

When this function is selected, the screen panel displayed depends on the data base file selected, as either an event number, a station name, a canister name, or a gauge name is requested and is used as a key to locate the appropriate data base records. In each case, the operations are the same:

- F1. enter
- F5. function
- F7. file
- F9. quit

The "function" and "file" operations are described in section 9.3.6.3.1; the "enter" and "quit" operations are described in section 9.3.6.1.

9.3.6.3.3.1. Event File.

A panel displaying an error message appears if the event is not located, and the user is allowed to enter another event number. When the records for the event to be accessed are located, the record containing data for the event is displayed. A smaller panel is displayed at the top of the screen to verify the user's command to delete the event record. The user is asked to respond by entering either a "Y" to delete the record, or an "N" to retain the record. It is important to note that deleting the event record also deletes all the event station records associated with the event in the event file; deleting the event record deletes all the records in the chain for that event. The operations to choose from are as follows:

- F1. next in chain

- F2. key change
- F3. function
- F4. file
- F9. quit

The "next in chain" operation allows the user to view the record of a station associated with the event. The record will appear on the screen and the small panel described above appears on the screen to verify the deletion command. By repeating this operation, the user can view every station record associated with the specified event and selectively delete event station records. The "key change" operation allows the user to return to the panel described above to enter another event name. The "function" and "file" operations are described in section 9.3.6.3.1; the "quit" operation is described in section 9.3.6.1.

9.3.6.3.3.2. Station File.

A panel displaying an error message appears if the station is not located, and the user is allowed to enter another station number. When the records for the station to be accessed are located, a record containing data for the station is displayed. A smaller panel is displayed at the top of the screen to verify the user's command to delete the station record. The user is asked to respond by entering either a "Y" to delete the record, or an "N" to retain the record. The operations to choose from are as follows:

- F1. next in chain
- F2. key change
- F3. function
- F4. file
- F9. quit

The "next in chain" operation allows the user to view another record of data for the specified station. The record will appear on the screen and the small panel described above appears on the screen to verify the deletion command. By repeating this operation, the user can view every station record associated with the specified station and selectively delete station records. The "key change" operation allows the user to return to the panel described above to enter another station name. The "function" and "file" operations are described in section 9.3.6.3.1; the "quit" operation is described in section 9.3.6.1.

9.3.6.3.3.3. Canister File.

A panel displaying an error message appears if the canister is not located, and the user is allowed to enter another canister number.

"ACCESS" USER MANUAL

When the records for the canister to be accessed are located, a record containing data for the canister is displayed. A smaller panel is displayed at the top of the screen to verify the user's command to delete the canister record. The user is asked to respond by entering either a "Y" to delete the record, or an "N" to retain the record. The operations to choose from are as follows:

- F1. next in chain
- F2. key change
- F3. function
- F4. file
- F9. quit

The "next in chain" operation allows the user to view another record of data for the specified canister. The record will appear on the screen and the small panel described above appears on the screen to verify the deletion command. By repeating this operation, the user can view every canister record associated with the specified canister and selectively delete canister records. "Key change" allows the user to return to the panel described above to enter another canister name. The "function" and "file" operations are described in section 9.3.6.3.1; the "quit" operation is described in section 9.3.6.1.

9.3.6.3.4.4. Gauge File.

A panel displaying an error message appears if the gauge is not located, and the user is allowed to enter another gauge number. When the records for the gauge to be accessed are located, a record containing data for the gauge is displayed. A smaller panel is displayed at the top of the screen to verify the user's command to delete the gauge record. The user is asked to respond by entering either a "Y" to delete the record, or an "N" to retain the record. The operations to choose from are as follows:

- F1. next in chain
- F2. key change
- F3. function
- F4. file
- F9. quit

The "next in chain" operation allows the user to view another record of data for the specified gauge. The record will appear on the screen and the small panel described above appears on the screen to verify the deletion command. By repeating this operation, the user can view every gauge record associated with the specified gauge and selectively delete gauge records. The "key change" operation allows the user to return to the panel

described above to enter another gauge name. The "function" and "file" operations are described in section 9.3.6.3.1; the "quit" operation is described in section 9.3.6.1.

9.3.6.3.4. Scan Records Function.

When this function is selected, the screen panel displayed depends on the data base file selected, as either an event number, a station name, a canister name, or a gauge name is requested and is used as a key to locate the appropriate data base records. In each case, the operations are the same.

- F1. enter
- F5. function
- F7. file
- F9. quit

The "function" and "file" operations are described in section 9.3.6.3.1; the "enter" and "quit" operations are described in section 9.3.6.1.

9.3.6.3.4.1. Event File.

A panel displaying an error message appears if the event is not located, and the user is allowed to enter another event number. When the records for the event to be accessed are located, the record containing data for the event is displayed. The operations to choose from are as follows:

- F1. next in chain
- F2. key change
- F3. function
- F4. file
- F9. quit

The "next in chain" operation allows the user to view the record of a station associated with the event. The record will appear on the screen. By repeating this operation, the user can view every station record associated with the specified event. The "key change" operation allows the user to return to the panel described above to enter another event name. The "function" and "file" operations are described in section 9.3.6.3.1; the "quit" operation is described in section 9.3.6.1.

9.3.6.3.4.2. Station File.

A panel displaying an error message appears if the station is not located, and the user is allowed to enter another station number. When the records for the station to be accessed are located, a record containing data for the station is displayed. The operations to choose from are as follows:

- F1. next in chain
- F2. key change
- F3. function
- F4. file
- F9. quit

The "next in chain" operation allows the user to view another record of data for the specified station. The record will appear on the screen. By repeating this operation, the user can view every station record associated with the specified station. The "key change" operation allows the user to return to the panel described above to enter another station name. The "function" and "file" operations are described in section 9.3.6.3.1; the "quit" operation is described in section 9.3.6.1.

9.3.6.3.4.3. Canister File.

A panel displaying an error message appears if the canister is not located, and the user is allowed to enter another canister number. When the records for the canister to be accessed are located, a record containing data for the canister is displayed. The operations to choose from are as follows:

- F1. next in chain
- F2. key change
- F3. function
- F4. file
- F9. quit

The "next in chain" operation allows the user to view another record of data for the specified canister. The record will appear on the screen. By repeating this operation, the user can view every canister record associated with the specified canister. The "key change" operation allows the user to return to the panel described above to enter another canister name. The "function" and "file" operations are described in section 9.3.6.3.1; the "quit" operation is described in section 9.3.6.1.

9.3.6.3.3.4. Gauge File.

A panel displaying an error message appears if the gauge is not located, and the user is allowed to enter another gauge number. When the records for the gauge to be accessed are located, a record containing data for the gauge is displayed. The operations to choose from are as follows:

- F1. next in chain
- F2. key change
- F3. function
- F4. file
- F9. quit

The "next in chain" operation allows the user to view another record of data for the specified gauge. The record will appear on the screen. By repeating this operation, the user can view every gauge record associated with the specified gauge. The "key change" operation allows the user to return to the panel described above to enter another gauge name. The "function" and "file" operations are described in section 9.3.6.3.1; the "quit" operation is described in section 9.3.6.1.

9.3.6.4. Program Termination.

Upon completion of the "ACCESS" program, the "CANSTR", "STATION", and "GAUGE" data base files must be made permanent to preserve any modifications made. This is explained in section 9.3.4.4.

"SCAN"
USER MANUAL

CHAPTER 10

"SCAN"
USER MANUAL

"SCAN"
USER MANUAL

10.1. GENERAL INFORMATION

10.1.1. Summary.

The "SCAN" program is designed to read a multiframe of channel data files and display the identifier (id) block from each channel in the file.

10.1.2. Environment.

This project is sponsored by the NNWSI project. Its development is the responsibility of the Ground Motion and Seismic Division (7111) of the Field Sciences Department (7110). Members of the project within Division 7111 are the sole users of the "SCAN" program.

The "SCAN" program is implemented on Sandia National Laboratories' CDC CYBER 180-855 model computer, located in the Central Computing Facility in Sandia's primary technical area, Area I, in building 800. The CYBER 180-855 operates in a dual state, which allows two operating systems to run concurrently. The "SCAN" program runs on the NOS/VE side of the CYBER 180-855. The current operating system version is 1.1.2 level 649.

10.1.3. References.

- 10.1.3.1. National Bureau of Standards, "Guidelines for Documentation of Computer Programs and Automated Data Systems, "Federal Information Processing Standards (FIPS) Publication #38, February, 1976.

10.2. APPLICATION

10.2.1. Description.

The "SCAN" program reads each data file in a multifile of channel data files and creates a listing of the id blocks from each of the channel files encountered in the multifile.

10.2.1.1. When to Use this Program.

The "SCAN" program should be used any time it is necessary to determine what data files are included in a multifile file, whether during processing or otherwise.

10.2.1.2. Program Functions.

This interactive program provides an output file that contains the id block from each data file within the multifile input file. The output file is in one of two formats. When run in normal mode, the first ten words of each id block are written to the output file. Alternately, the user may specify that a full id block dump be provided, and all 100 words of each id block are written to the output file. The output file is displayed on the terminal screen and disposed to the plotter to generate a hard-copy listing. It is also accessible at program termination, as it resides in the local file space.

10.2.2. Operation.

Only those members of the Ground Motion and Seismic Division (7111) who are working on the project and have need to execute the "SCAN" program are authorized to access the procedure that directs the program execution. At the present time, there are three such users.

10.2.3. Equipment.

The "SCAN" program resides on the NOS/VE side of the dual state CDC CYBER 180-855. The CYBER 180-855, under NOS/VE, is configured with 6.9 million words (55.2 million bytes) of memory, and three 895 disks, each with 4 spindles, for a total of 3.6 giga-bytes of online disk space.

10.2.4. Structure.

All software for the "SCAN" program is written in FORTRAN 5 and executed under the control of CDC's NOS/VE.

The "SCAN" program is written in a modular fashion. Each routine performs one basic function or set of related functions, thereby providing a program that is easy to read and modify. The main program

"SCAN" USER MANUAL

interprets the user's selection and directs the appropriate routine to verify and process the input data. When the user's input has been dealt with appropriately, the main program determines the functions to be performed and initiates the proper routine for processing.

10.2.5. Performance.

Currently, the scan program exists and is run interactively under control of the NOS/VE on the dual state CYBER 180-855 computer. The program can be run in batch mode if desired. Program execution is invoked from a procedure. The procedure can easily be modified to start the program in batch mode rather than in interactive mode. No changes need be made to the program to run in batch mode.

10.2.6. Inputs, Processing, and Outputs.

10.2.6.1. Inputs.

The "SCAN" program requires as input only two parameters: the name of the data file to be scanned, and the type of scan output to be created, as described in section 10.2.1.2.

10.2.6.2. Processing.

10.2.6.2.1. Read the Parameter List.

The program reads the input parameter values from the parameter file created by the "RUN_SCAN" procedure from the parameter values specified on the procedure call statement. When a value for an optional parameter is not specified in the procedure call statement, the program uses the default value for the parameter.

10.2.6.2.2. Create the Scan List.

The input data file is read from the beginning-of-file position. An identifier block is read from the data file and written to the output listing file in the specified format. If there are any modification records in the input file, they are read, and in essence, skipped. The data records are treated as the modification records, and are skipped. The input file is positioned at the beginning of the next data file. The program continues reading and skipping the records in the input data file, always writing the id block to the output listing file in the specified format. The program terminates normally when an end-of-information is encountered rather than an identifier block for another data file.

"SCAN"
USER MANUAL

10.2.6.3. Outputs.

Upon completion of the "SCAN" program, the output file containing the id blocks of each of the data files within the input data file is in the local file space of the user. This file will also contain any messages issued by the program during execution. The file can be made permanent by the user for future reference.

10.3. PROCEDURES AND REQUIREMENTS

10.3.1. Initiation.

10.3.1.1. The Procedure.

The "SCAN" program is initiated and executed under the control of a procedure named "RUN_SCAN." "RUN_SCAN" performs all of the commands necessary to execute the "SCAN" program, which simplifies the user's task of running the program. The parameters that comprise the parameter input file for the program are required on the procedure call statement. With these given values, the procedure generates the parameter input file that is used by the program.

10.3.1.2. Getting a Copy of the Procedure.

Both the "RUN_SCAN" procedure and the "SCAN" program source are resident in disk space allocated to a particular member of the Ground Motion and Seismic Division (7111). This user's disk space is called a "catalog" in NOS/VE. For a user other than the "owner" to access the procedure, the user name of the owner must be specified on the command that retrieves the file. The following describes the steps necessary for a user to get a copy of the procedure:

```
COPY_FILE .MLSANDE.COMMAND_CATALOG.RUN_SCAN RUN_SCAN
```

The "COPY_FILE" command will work properly only if the procedure file has been made available for access by the user requesting it. This will make a copy of the procedure in the user's catalog and will name it "RUN_SCAN." When the procedure is executed, it will get copies of the files necessary to execute the program, including the program source.

10.3.1.3. Invoking the Procedure.

"RUN_SCAN" creates a parameter input file for the program from the parameter list specified on the statement that invokes the procedure. The "RUN_SCAN" procedure will accept two parameters: the name of the input data file containing data to be scanned and the type of scan output to be created. The name of the input data file is a required parameter. The default value for the type of scan to be produced is used if the parameter is not entered. The parameters can be specified positionally or by parameter name, and must be separated by a space or a comma. Parameters specified by a name have the following form:

```
name=value
```

where the name is the parameter name and value is the parameter value.
This is the command that invokes the procedure:

```
RUN_SCAN DF=data_file_name ST=scan_type
```

10.3.1.4. Compilation and Execution.

The procedure issues the command to compile the program source. When the source does not compile normally, a listing of the errors is displayed on the screen and the procedure is terminated. When the source compiles normally, the program is executed.

10.3.2. Input.

10.3.2.1. Requirements.

10.3.2.1.1. Origin.

The input multifile is generated in several ways, primarily by the program that reads the data from the tape after digitization. Multifile files are also generated by the processing programs.

10.3.2.1.2. Medium.

The input data file is available to the program as it exists in a file on disk.

10.3.2.1.3. Restrictions.

There are no security restrictions placed on the operation of the "SCAN" program, as it resides in the open-partition CYBER 180-855 and all data is considered unclassified. Execution of the program and access to the input files is limited to only those users having knowledge about the program, the potential input files, and the user's catalog in which the files reside.

10.3.2.2. Input Formats.

10.3.2.2.1. Length.

The name of the input file may be a maximum of 14 characters in length and is not enclosed in any quote marks. The parameter indicating the type of scan to be produced may be a maximum of 4 characters in length and is not enclosed in any quote marks.

10.3.2.2.2. Format.

The input file is a multifile file, meaning that it is composed of one or more files, and all data files within the multifile are of the same format. The data within one of these files can logically be divided into three sections: the identifier block, the modification records, and the data records. Refer to section 10.3.3. for a detailed description of the input file. The parameter name that indicates the type of scan to generate must be either "NORM" or "ALL". "NORM" is specified to create a 10-word identifier block for each data file read, and "ALL" is specified to create a 100-word identifier block. When no scan format type is specified on the procedure call statement, the default value "NORM" is used.

10.3.2.2.3. Labels.

The parameters can be specified positionally or by parameter name, and must be separated by a space or a comma. Refer to section 10.3.1.3. for a description of the labels for the input parameters.

10.3.2.2.4. Sequence.

When the parameter values are specified without parameter names, the values must be entered in the appropriate sequence described in section 10.3.1.3. When parameter names are used with the parameter values, the order of specification on the procedure call statement is meaningless.

10.3.2.2.5. Combination.

Any combination of upper and lower case characters may be entered, as the procedure interprets all characters as their upper case equivalents. Some parameter values may require a combination of characters and numbers; either upper or lower case may be entered for the character parts of these parameter values.

10.3.3. Sample Inputs.

10.3.3.1. The Input File.

The input multifile must be of a specific format. This format is the natural result of reading the digitized data from tape, which is performed prior to executing this program. The format is briefly described here.

10.3.3.1.1. The Identifier Block.

The first 100-words (10 80-character lines) of a data file is the identifier (ID) block. It gives vital information regarding the data records in the file and to what event they pertain.

10.3.3.1.2. The Modification Records.

The modification (MOD) records are written to a data file by the data processing programs as the file proceeds through the data processing system. The MOD records contain information specific to the various processes performed on the data. A value in one of the words of the ID block indicates how many MOD records have been written to the data file. A MOD record, however, consists of three 80-character records. The value from the ID block must be multiplied by three to determine the correct number of MOD records.

10.3.3.1.3. The Data Records.

The data records contain the data in integer format. There are 16 data values per line, or record, of the data file. A value in one of the words of the id block indicates how many data values are in the data file. This value is divided by 16 to determine the number of complete records of data as well as the number of data values in the last record of the file.

10.3.3.2. The Procedure Call Statement.

The following are examples of initiating the "SCAN" program.

```
RUN_SCAN input_file NORM
```

```
RUN_SCAN input_file
```

```
Run_scan st=norm df=input_file
```

10.3.4. Output.

10.3.4.1. Use.

The scan output is used during processing to determine what data files are contained in a multifile. Scans are also generated before a multifile is written to tape, and thus the scans are later used to determine what files are available on a particular tape.

10.3.4.2. Frequency.

The scan output is generated as often as necessary.

10.3.4.3. Variations.

The format of the scan output will vary depending on the type of scan requested on the procedure call. As stated in section 10.2.1.2, the first ten words of each id block for each file in the input multifile can comprise the output file. As an alternative, all 100 words of each id block are written to and comprise the output file.

10.3.4.4. Destination/Disposition.

The hard-copy scan output file is disposed to the line printer available for receiving output from the CYBER 180-855. It is located in the Central Computing Facility in Sandia's primary technical area, Area 1, in building 800. The listings are delivered by courier to room 166 in building 806 on a regular daily basis, and are used while processing the data.

10.3.4.5. Medium.

The "SCAN" program generates a hard-copy listing of each scan performed.

10.3.4.6. Sample Outputs.

Figure 10.1. represents a sample output from the "SCAN" program.

10.3.4.6.1. Definition.

The headings on the scan output, as seen in Figure 10.1, identify the various words in a data file id block. "FILE NO." indicates the position in the multifile of a particular data file. "DATA ID" lists the first word from the id block which indicates whether the data is in raw, rotated, or filtered format. The remaining headings refer to the appropriate words from the id block: "STATION" indicates the name of the station for which data is in the file; "ORIENT" indicates the orientation of the gauge; "EVENT" indicates the name of the event for which the data pertains; "ROTATE" indicates the rotation angle of the gauge, after the data has been processed by the "ROTATE" program; "AMP" indicates the amplification of the gauge used to collect the data; "TRACK" indicates the track number on the tape on which the data was recorded; "FREQ" indicates the frequency on which the data is recorded on the track; "START" indicates the start time of the data; "END" indicates the stop time of the data.

"SCAN"
USER MANUAL

Figure 10.1

SCAN OF TAPE scan_data

FILE NO.	DATA ID	STATION	CRIENT	EVENT	ROTATE	AMP	TR FREQ	START	END
1	RAW	1 W-30 TOP	AV	LABQUARK		830.00	TR1070.0	-60.0000	119.9980
2	RAW	1 W-30 TOP	A-270.	LABQUARK		830.00	TR1052.5	-60.0000	119.9980
3	RAW	1 W-30 TOP	A-0.	LABQUARK		830.00	TR1040.0	-60.0000	119.9980

THIS DATA WAS PROCESSED IN ACCORDANCE WITH ETP-7111-1.

SCAN - NORMAL TERMINATION.
3 FILES PROCESSED.

10.3.4.6.2. Source.

All values listed in the scan output are taken directly from the id blocks of the files in the input multifile, with the exception of the file number, which is generated by the program.

10.3.4.7. At Program Termination.

The program terminates normally when an end-of-information is encountered rather than an identifier block for a data file. Any identifier blocks that the program reads before it terminates are displayed and retained in the output file. It should be noted that the output file exists in the local file space, a hard-copy listing is generated, and if it is necessary to retain a copy in disk space, the user must issue the command to make it a permanent file.

10.3.5. Error and Recovery.

If, while reading the input data file, an end-of-information is encountered anywhere other than at the beginning of the assumed next data file, the program terminates abnormally. The program and the procedure both issue error messages that are displayed on the screen, and the procedure is terminated.

There are no special restart procedures. The procedure "RUN_SCAN" performs several steps at the completion of the program that ensure successful termination of the procedure. This allows multiple restarts of the procedure, regardless of program termination status.

"PLOT"
USER MANUAL

CHAPTER 11

"PLOT"
USER MANUAL

11.1. GENERAL INFORMATION

11.1.1. Summary.

The "PLOT" program is a general plotting program used to plot data in its raw format, as it is prior to any processing, and to generate specialized plots from the data at various stages of processing.

11.1.2. Environment.

This project is sponsored by the NNWSI project. Its development is the responsibility of the Ground Motion and Seismic Division (7111) of the Field Sciences Department (7110). Members of the project within Division 7111 are the sole users of the "PLOT" program.

The "PLOT" program is implemented on Sandia National Laboratories' CDC CYBER 180-855 model computer, located in the Central Computing Facility in Sandia's primary technical area, Area I, in building 800. The CYBER 180-855 operates in a dual state, which allows two operating systems to run concurrently. The "PLOT" program runs on the NOS/VE side of the CYBER 180-855. The current operating system version is 1.1.2 level 649.

11.1.3. References.

11.1.3.1. National Bureau of Standards, "Guidelines for Documentation of Computer Programs and Automated Data Systems, "Federal Information Processing Standards (FIPS) Publication #38, February, 1976.

11.1.3.2. Integrated Software Systems Corporation, "Display Integrated Software System and Plotting LAnguage" software library, current version 9.2, 1505 Sorrento Valley Rd. San Diego, California 92121.

11.2. APPLICATION

11.2.1. Description.

The "PLOT" program reads an input multiframe of channel data files and generates a plot for each of the data files. The plots produced are of a predetermined format, however, the format can be modified by parameters specified when program execution is invoked.

11.2.1.1. When to Use this Program.

The "PLOT" program should be used to plot the data in its raw format prior to any processing. It should also be used to produce plots of the filtered, rotated, and vectored data in an expanded format; that is, plots that comprise more than a single plot frame.

11.2.1.2. Program Functions.

Several parameters that specify certain plot characteristics can be specified when the "PLOT" program is executed. This provides a program that can be tailored to generate plots of specific formats. Any, all, or none of the following parameters can be specified when the plot program is executed:

1. The number of plot frames that compose a plot.
2. The smoothness of the curve, or spline, to be used when generating the plot.
3. The start time for the data segment to be plotted.
4. The stop time for the data segment to be plotted.
5. The device to which the plots are to be disposed.

11.2.2. Operation.

Only those members of the Ground Motion and Seismic Division (7111) who are working on the project and have need to execute the "PLOT" program are authorized to access the procedure that directs the program execution. At the present time, there are three such users.

11.2.3. Equipment.

The "PLOT" program resides on the NOS/VE side of the dual state CDC CYBER 180-855. The CYBER 180-855, under NOS/VE, is configured with 6.9 million words (55.2 million bytes) of memory, and three 895 disks, each with 4 spindles, for a total of 3.6 giga-bytes of online disk space.

11.2.4. Structure.

Software for the "PLOT" program consists of FORTRAN 5 source code with calls to plot routines in a software library called "DISSPLA", both of which are resident on and executed under the control of CDC's NOS/VE.

The "PLOT" program is written in a modular fashion. Each routine performs one basic function or set of related functions, thereby providing a program that is easy to read and modify. The main program interprets the user's selection and directs the appropriate routine to verify and process the input data. When the user's input has been dealt with appropriately, the main program determines the functions to be performed and initiates the proper routine for processing.

11.2.5. Performance.

Currently, the plot program exists and is run interactively under control of the NOS/VE on the dual state CYBER 180-855 computer. The program can be run in batch mode if desired. Program execution is invoked from a procedure. The procedure can easily be modified to start the program in batch mode rather than in interactive mode. No changes need be made to the program to run in batch mode.

11.2.6. Inputs, Processing, and Outputs.

11.2.6.1. Inputs.

Six parameters can be specified for the "PLOT" procedure: the name of the input data file, the number of frames for a plot, the smoothness of the curve, the start time of the plot, the stop time of the plot, and the plot device parameter. The name of the input data file is the only required parameter, however.

11.2.6.1.1. Description.

11.2.6.1.1.1. Input File Name.

The input file must be available as a disk file in the NOS/VE system. The input file name must be specified on the procedure call statement.

11.2.6.1.1.2. Number of Frames for a Plot.

In some cases, it is desired to plot a segment of data on more than one page, thus producing an "expanded" plot. The number of frames for a plot determines whether to produce expanded plots. The minimum number of frames for a plot is 1 and the maximum is 4. If this parameter is not specified on the procedure call statement, the program uses the default value of 1.

11.2.6.1.1.3. Smoothness of the Curve.

In almost all cases, a smooth curve should be generated on the plots. A value of 1 produces the smoothest curve and a value of 4 produces a straight line curve. In almost all cases, a straight line curve should be generated on the plots. This parameter should rarely, if ever, be changed from the value of 4, which is the default value used when the parameter is not specified on the procedure call statement.

11.2.6.1.1.4. Start Time of the Plot.

It may be desired to plot only a portion of a data file. When the start time is specified on the procedure call statement, it must be given in seconds; half-seconds can be specified, as in "-20.5". When this parameter is not specified on the procedure call statement, the program will use the start time indicated in the data.

11.2.6.1.1.5. Stop Time of the Plot.

It may be desired to plot only a portion of a data file. When the stop time is specified on the procedure call statement, it must be given in seconds; half-seconds can be specified, as in "85.5". When this parameter is not specified on the procedure call statement, the program will use the stop time indicated in the data.

11.2.6.1.1.6. Plot Device.

When this parameter is specified on the procedure call statement, it must be a three-character code recognized by the NOS/VE system as a graphics output device. When this parameter is not specified, it is set to "HC1," which disposes the plot output to the Versatec plotter.

11.2.6.2. Processing.

11.2.6.2.1. Read the Parameter List.

The program reads the input parameter values from the parameter file created by the "RUN PLOT" procedure from the parameter values specified on the procedure call statement. When a value for an optional parameter is not specified in the procedure call statement, the program uses the default value for the parameter.

11.2.6.2.2. Create The Plot File.

The program creates a plot file containing the plots for each of the data files in the input file. When the terminal is specified as the plot device in the "RUN PLOT" procedure, each plot is generated on the terminal. When the plots are to be disposed to the plotter, nothing appears on the terminal screen until program termination. The input data file is read from the beginning-of-file position. An identifier block is read from the data file and the following values are extracted: the number of data values in the file, the number of modification records in the file, the delta time, the data start time, the data stop time, and the multiplier to reconstruct the data. The modification records are read and skipped. The data is then read from the data file and expanded from its integer format to real format. If start and stop times indicating a data segment to be plotted were specified in the parameter list, the data segment is extracted and the plot is generated. Otherwise, the plot is generated using the entire data set.

The program reads each data file from the input multifile and generates plots based on the start and stop times, whether specified in the parameter list or read from the id blocks in the data files.

11.2.6.3. Outputs.

Upon completion of the "PLOT" program, the plot file containing the plots for each of the channel files in the input multifile is disposed to the plotter to generate hard-copy plots. The command that transmits the plot file to the NOS operating system for disposition to the plotter is also displayed with all of its associated subcommands. If the program was initiated to generate the plots on the terminal, each plot is generated on the terminal screen.

11.3. PROCEDURES AND REQUIREMENTS

11.3.1. Initiation.

11.3.1.1. The Procedure.

The "PLOT" program is initiated and executed under the control of a procedure named "RUN_PLOT." "RUN_PLOT" performs all of the commands necessary to execute the "PLOT" program and dispose the plot file to the specified output device, which simplifies the user's task of running the program. The parameters that comprise the parameter input file for the program are specified on the procedure call statement. With these given values, the procedure generates the parameter input file that is used by the program, and also issues the command to dispose the plot file to the specified device.

11.3.1.2. Getting a Copy of the Procedure.

Both "RUN_PLOT" and the program source "PLOT" are resident in disk space allocated to a particular member of the Ground Motion and Seismic Division (7111). For a user other than the owner to access the procedure, the user name of the owner must be specified on the command that retrieves the file. The following command allows the user to get a copy of the procedure:

```
COPY_FILE .MLSANDE.COMMAND_CATALOG.RUN_PLOT RUN_PLOT
```

The "COPY_FILE" command will work properly only if the procedure file has been made available for access by the user requesting it. This command will make a copy of the procedure and will name it "RUN_PLOT." When the procedure is executed, it will access all of the files necessary to execute the program.

11.3.1.3. Invoking the Procedure.

"RUN_PLOT" creates a parameter input file for the program from the parameter list specified on the statement that invokes the procedure. The "RUN_PLOT" procedure will accept six parameters: the name of the input data file, the start time of the data segment to plot, the stop time of the data segment to plot, the number of frames for a plot, the degree of curve rounding to use, and the plot device to receive the plots. The parameters can be specified positionally or by parameter name, and must be separated by a space or a comma. Parameters specified by a name have the following form:

```
name=value
```

where name is the parameter name and value is the parameter value.
This is the command that invokes the procedure:

```
RUN_PLOT F=data_file ST=start_time ET=end_time NF=number_of_frames  
DN=device_name S=spline
```

11.3.1.4. Compilation and Execution.

The procedure compiles the program source. When the source does not compile normally, a listing of the errors is displayed on the screen and the procedure is terminated. When the source compiles normally, program execution is initiated.

11.3.2. Input.

11.3.2.1. Procedure Parameters.

The "PLOT" procedure will accept the six parameters described in section 11.3.1.2.

The procedure creates a parameter input file for the program from the parameter values specified on the procedure call statement. The name of the input data file is the only required parameter. The default value for a parameter is used if an optional parameter is not entered. Refer to section 11.2.6.1. for a description of the parameters and their default values. The parameter names need not be included when the parameter values are given positionally as shown above in section 11.3.1.3.

11.3.2.2. Requirements.

11.3.2.2.1. Frequency.

The "PLOT" program should be run to generate plots of the raw data prior to any processing. "PLOT" is also run during processing to generate expanded plots. The appropriate procedures that initiate the processing programs also initiate the execution of the "PLOT" program.

11.3.2.2.2. Origin.

The input multifile is generated in several ways. The data can be created by the program that reads the data from the tape after digitization, or it can be the resulting data file after executing one of the processing programs. The program parameters are selected by the user when the procedure is initiated.

11.3.2.2.3. Medium.

The input data file is available to the program as it exists in a file on disk. The program parameters are specified on the command line entered on the terminal.

11.3.2.2.4. Restrictions.

There are no security restrictions placed on the operation of the "PLOT" program, as it resides in the open-partition CYBER 180-855 and all data is considered unclassified. Execution of the program and access to the input files is limited to only those users having knowledge about the program, the potential input files, and the user's catalog in which the files reside.

11.3.2.2. Input Formats.

11.3.2.2.1. Length.

The name of the input file may be a maximum of 14 characters in length and is not enclosed in any quote marks.

11.3.2.2.2. Format.

The input file is a multifile file, and all data files within the multifile are of the same format. Each of the data files can logically be divided into three sections: the identifier block, the modification records, and the data records. Refer to section 11.3.3.1 for a detailed description of a data file.

11.3.2.2.3. Labels.

The parameter values can be specified positionally or by parameter name, and must be separated by a space or a comma. Refer to section 11.3.1.3 for a description of the labels for the input parameters.

11.3.2.2.4. Sequence.

When the parameter values are specified without parameter names, the values must be entered in the appropriate sequence described in section 11.3.1.3. When parameter names are used with the parameter values, the order of specification on the procedure call statement is meaningless.

11.3.3. Sample Inputs.

11.3.3.1. The Input File.

The input multifile must be of a specific format. This format is the natural result of reading the digitized data from tape, as well as executing one of the processing programs. The format is briefly described here.

11.3.3.1.1. The Identifier Block.

The first 100-words (10 80-character lines) of a data file is the identifier (ID) block. It gives vital information regarding the data records in the file and to what event they pertain.

11.3.3.1.2. The Modification Records.

The modification (MOD) records are written to a data file by the data processing programs as the file proceeds through the data processing system. The MOD records contain information specific to the various processes performed on the data. A value in one of the words of the ID block indicates how many mod records have been written to the data file. A MOD record, however, consists of three 80-character records. The value from the ID block must be multiplied by three to determine the correct number of MOD records.

11.3.3.1.3. The Data Records.

The data records contain the data in integer format. There are 16 data values per line, or record, of the data file. A value in one of the words of the ID block indicates how many data values are in the data file. This value is divided by 16 to determine the number of complete records of data as well as the number of data values in the last record of the file.

11.3.3.2. The Procedure Call Statement.

The following are examples of initiating the "PLOT" program.

```
RUN_PLOT f=data_file st=-20 et=85 s=2
```

```
RUN_plot data_file
```

```
Run_plot data_file NF=3
```

"PLOT"
USER MANUAL

11.3.4. Output.

11.3.4.1. Use.

The plot output is retained and stored according to event name by Division 7111.

11.3.4.2. Variations.

The format of the plots generated by the plot program will vary depending on the type of plot requested. When expanded plots are produced, the data for a single channel is plotted on several plot frames over several pages. Otherwise, the data for a single channel is plotted on a single plot frame.

11.3.4.3. Destination.

11.3.4.3.1. Line Printer.

There is only one line printer available for receiving output directly from the CYBER 180-855. It is located in the Central Computing Facility in Sandia's primary technical area, Area I, in building 800. The procedure that executes this program directs all output to this line printer.

11.3.4.3.2. Plotter.

The output generated from the "PLOT" program is generally disposed to a Versatec plotter. There are several plotters available for use. One is located in the Central Computing Facility in Sandia's primary technical area, Area I, in building 800. Another, and the one to which all plots produced by this program are disposed, is located in Area I, building 806, room 166.

11.3.4.3.3. Terminal.

This program can also dispose the plot output to the terminal running the program, providing the terminal is capable of producing graphics and is supported as a graphics device. This is achieved by specifying a parameter in the procedure that executes the program.

11.3.4.4. Medium.

The plots are stored in hard-copy format, as generated by a Versatec plotter.

"PLOT"
USER MANUAL

11.3.4.5. Disposition.

The plots generated by the "PLOT" program are stored by Division 7111 personnel in Area 1, building 806.

11.3.4.6 Sample Outputs.

Figures 11.1, 11.2, and 11.3 represent sample outputs for a station generated by the "PLOT" program.

Figure 11.1

Figure 11.2

RMP X 830.00
TRACK 10
52.5 KHZ
1987-03-13
10:25:31

DEL: .002
MIN: -.3037E-01
MAX: .3140E-01
AZM: 194.6659
RNG: 55362.46
VSN:

MUL: .1885E-05

Figure 11.3

AMP X 830.00
TRACK 10
40.0 KHZ
1987-03-13
10:26:26

DEL: .002
MIN: -.2993E-01
MAX: .2698E-01
AZM: 194.6659
RNG: 55362.46
VSN:

MUL: .1737E-05

11.3.4.7. At Program Termination.

The program terminates normally when an end-of-information is encountered rather than an identifier block for a data file. When the "PLOT" program terminates normally, and the plots are to be disposed to the plotter, a message indicating the number of plots generated is displayed on the terminal. The command that transmits the plot file to the NOS operating system for disposition to the plotter is also displayed with all of its associated subcommands. When the terminal is specified as the plot device in the "RUN_PLOT" procedure, each plot is generated on the terminal.

11.3.5. Error and Recovery.

There are no special restart procedures. The procedure ensures its own successful termination. This allows multiple restarts of the procedure, regardless of program termination status.

If, while reading the input data file, an end-of-information is encountered anywhere other than at the beginning of the assumed next data file, the program terminates abnormally. The program and the procedure both issue error messages that are displayed on the screen, and the procedure is terminated.

"SPECTRA"
USER MANUAL

CHAPTER 12
"SPECTRA"
USER MANUAL

"SPECTRA"
USER MANUAL

12.1. GENERAL INFORMATION

12.1.1. Summary.

The "SPECTRA" program calculates and plots power density spectra for the noise and signal-and-noise segments of a data file.

12.1.2. Environment.

This project is sponsored by the NNWSI project. Its development is the responsibility of the Ground Motion and Seismic Division (7111) of the Field Sciences Department (7110). Members of the project within Division 7111 are the sole users of the "SPECTRA" program.

The "SPECTRA" program is implemented on Sandia National Laboratories' CDC CYBER 180-855 model computer, located in the Central Computing Facility in Sandia's primary technical area, Area I, in building 800. The CYBER 180-855 operates in a dual state, which allows two operating systems to run concurrently. The "SPECTRA" program runs on the NOS/VE side of the CYBER 180-855. The current operating system version is 1.1.2 level 649.

12.1.3. References.

12.1.3.1. National Bureau of Standards, "Guidelines for Documentation of Computer Programs and Automated Data Systems, "Federal Information Processing Standards (FIPS) Publication #38, February, 1976.

12.1.3.2. Integrated Software Systems Corporation, "Display Integrated Software System and Plotting LAnguage" software library, current version 9.2, 1505 Sorrento Valley Rd. San Diego, California 92121.

"SPECTRA" USER MANUAL

12.2. APPLICATION

12.2.1. Description.

The "SPECTRA" program calculates and plots a power density spectrum for each channel of data in an input multifile. The method used for spectral estimation is that of average periodograms.

12.2.1.1. When to Use this Program.

The "SPECTRA" program should be used to determine the high pass and low pass cutoff frequencies prior to executing the filtering program.

12.2.1.2. Program Functions.

Several parameters that specify certain characteristics of the data can be specified when the "SPECTRA" program is executed. This provides a program that can vary the amount of data to use in calculating the power density spectra. Any, all, or none of the following parameters can be specified when the plot program is executed:

1. Noise start time.
2. Noise end time.
3. Signal start time.
4. Signal end time.
5. Block size to use in the calculation.

12.2.2. Operation.

Only those members of the Ground Motion and Seismic Division (7111) who are working on the project and have need to execute the "SPECTRA" program are authorized to access the procedure that directs the program execution. At the present time, there are three such users.

12.2.3. Equipment.

The "SPECTRA" program resides on the NOS/VE side of the dual state CDC CYBER 180-855. The CYBER 180-855, under NOS/VE, is configured with 6.9 million words (55.2 million bytes) of memory, and three 895 disks, each with 4 spindles, for a total of 3.6 giga-bytes of online disk space.

"SPECTRA" USER MANUAL

12.2.4. Structure.

Software for the "SPECTRA" program consists of FORTRAN 5 source code with calls to plot routines in a software library called "DISSPLA", both of which are resident on and executed under the control of CDC's NOS/VE.

The "SPECTRA" program is written in a modular fashion. Each routine performs one basic function or set of related functions, thereby providing a program that is easy to read and modify. The main program interprets the user's selection and directs the appropriate routine to verify and process the input data. When the user's input has been dealt with appropriately, the main program determines the functions to be performed and initiates the proper routine for processing.

12.2.5. Performance.

Currently, the spectrum program exists and is run interactively under control of the NOS/VE on the dual state CYBER 180-855 computer. The program can be run in batch mode if desired. Program execution is invoked from a procedure. The procedure can easily be modified to start the program in batch mode rather than in interactive mode. No changes need be made to the program to run in batch mode.

12.2.6. Inputs, Processing, and Outputs.

12.2.6.1. Inputs.

Seven parameters can be specified for the "SPECTRA" procedure: the input file name, the noise start time, the noise end time, the signal start time, the signal end time, the block size, and the plot device parameter. The name of the input data file is the only required parameter, however.

12.2.6.1.1. Description.

12.2.6.1.1.1. Input File Name.

The input file must be available as a disk file in the NOS/VE system. The input file name must be specified on the procedure call statement.

12.2.6.1.1.2. The Start Time of the Noise Segment.

The noise start time must be specified in seconds; half-seconds can be specified, as in "-55.5." This parameter is not required, and when a value for it is not specified in the procedure, the program uses a default value of -60.0 seconds.

12.2.6.1.1.3. The End Time of the Noise Segment.

The noise end time must be specified in seconds; half-seconds can be specified, as in "5.5." This parameter is not required, and when a value for it is not specified in the procedure, the program uses a default value of 0.0 seconds.

12.2.6.1.1.4. The Start Time of the Signal Segment.

The start time must be specified in seconds; half-seconds can be specified, as in "5.5." This parameter is not required, and when a value for it is not specified in the procedure, the program uses a default value of 0.0 seconds.

12.2.6.1.1.5. The End Time of the Signal Segment.

The end time must be specified in seconds; half-seconds can be specified, as in "95.5." This parameter is not required, and when a value for it is not specified in the procedure, the program uses a default value of 90.0 seconds.

12.2.6.1.1.6. The Block Size.

The block size must be an integer value. This parameter is not required, and when a value for it is not specified in the procedure, the program uses a default value of 0, which results in an actual block size of 2048.

12.2.6.1.1.7. Plot Device.

When this parameter is specified on the procedure call statement, it must be a three-character code recognized by the NOS/VE system as a graphics output device. When this parameter is not specified, it is set to "HC1", which disposes the plot output to the Versatec plotter.

12.2.6.2. Processing.

12.2.6.2.1. Read the Parameter List.

The program reads the input parameter values from the parameter file created by the "RUN_SPECTRA" procedure from the parameter values specified on the procedure call statement. When a value for an optional parameter is not specified in the procedure call statement, the program uses the default value for the parameter.

12.2.6.2.2. Calculate the Spectra.

The input data file is read from the beginning-of-file position. An identifier block is read from the data file and the following values are extracted: the number of data values in the file, the number of modification records in the file, the delta time, the data start time, the data stop time, the minimum data value, the maximum data value, and the multiplier to reconstruct the data. The modification records are read and skipped. The data is then read from the data file and expanded from its integer format to real format. Using either the start and end times specified by the user in the procedure or by their default values, the indices for the start and end times for the noise and the signal segments of the data file are calculated.

12.2.6.2.2.1. Calculate the Periodograms.

Periodograms for the noise and signal segments of the data are calculated based on the data segment defined by the start and end times. The data segment, or vector, is divided into half-overlapping blocks of a given length. This length is the block size (ref: section 12.4.2.6.). Each block is normalized using the average of the first hundred points of the vector. A data window with a root mean square (RMS) of 1 is then applied to each block and a fast Fourier transform is performed. The squared magnitudes of the amplitudes at each frequency are accumulated over all the blocks and then scaled to give the estimate of the power density spectrum.

12.2.6.2.3. The Spectra Plot.

The estimates for both the power density spectrum for the noise segment and the power density spectrum for the noise and data segment are written to the plot file.

A similar plot is generated for each of the data files in the input multifile.

12.2.6.3. Outputs.

Upon completion of the "SPECTRA" program, the plot file containing the spectrum plots for each of the channel files in the input multifile is disposed to the plotter to generate hard-copy plots. The command that transmits the plot file to the NOS operating system for disposition to the plotter is also displayed with all of its associated subcommands. If the program was initiated to generate the plots on the terminal, each plot is generated on the terminal screen.

12.3. PROCEDURES AND REQUIREMENTS

12.3.1. Initiation.

12.3.1.1. The Procedure.

The "SPECTRA" program is initiated and executed under the control of a procedure named "RUN_SPECTRA." "RUN_SPECTRA" performs all of the commands necessary to execute the "SPECTRA" program and dispose the plot file to the specified output device, which simplifies the user's task of running the program. The parameters that comprise the parameter input file for the program are specified on the procedure call statement. With these given values, the procedure generates the parameter input file that is used by the program, and also issues the command to dispose the plot file to the specified device.

12.3.1.2. Getting a Copy of the Procedure.

Both "RUN_SPECTRA" and the program source "SPECTRA" are resident in disk space allocated to a particular member of the Ground Motion and Seismic Division (7111). For a user other than the owner to access the procedure, the user name of the owner must be specified on the command that retrieves the file. The following command allows the user to get a copy of the procedure:

```
MLSANDE.COMMAND_CATALOG.RUN_SPECTRA RUN_SPECTRA
```

The "COPY_FILE" command will work properly only if the procedure file has been made available for access by the user requesting it. This command will make a copy of the procedure and will name it "RUN_SPECTRA." When the procedure is executed, it will access all of the files necessary to execute the program.

12.3.1.3. Invoking the Procedure.

"RUN_SPECTRA" creates a parameter input file for the program from the parameter list specified on the statement that invokes the procedure. The "RUN_SPECTRA" procedure will accept seven parameters: the name of the input data file, the noise start time, the noise end time, the signal start time, the signal end time, the block size, and the name of plot device to receive the plots. The parameters can be specified positionally or by parameter name, and must be separated by a space or a comma. Parameters specified by a name have the following form:

```
name=value
```

where name is the parameter name and value is the parameter value. This is the command that invokes the procedure:

"SPECTRA" USER MANUAL

```
RUN_SPECTRA DF=data_file NST=noise_start_time NET=noise_end_time..  
             SST=signal_start_time SET=signal_end_time..  
             BS=block_size DN=device_name
```

12.3.1.4. Compilation and Execution.

The procedure compiles the program source. When the source does not compile normally, a listing of the errors is displayed on the screen and the procedure is terminated. When the source compiles normally, program execution is initiated.

12.3.2. Input.

12.3.2.1. Procedure Parameters.

The "RUN_SPECTRA" procedure will accept the seven parameters described in section 12.3.1.3.

The procedure creates a parameter input file for the program from the parameter values specified on the procedure call statement. The name of the input data file is the only required parameter. The default value for a parameter is used if an optional parameter is not entered. The parameter names need not be included when the parameter values are given positionally as shown above in section 12.3.1.3.

12.3.2.2. Requirements.

12.3.2.2.1. Frequency.

The "SPECTRA" program should be run to generate plots of the raw data prior to any processing.

12.3.2.2.2. Origin.

The input multifile is generated by the program that reads the data from the tape after digitization. The program parameters are selected by the user when the procedure is initiated.

12.3.2.2.3. Medium.

The input data file is available to the program as it exists in a file on disk. The program parameters are specified on the command line entered on the terminal.

12.3.2.2.4. Restrictions.

There are no security restrictions placed on the operation of the "SPECTRA" program, as it resides in the open-partition CYBER 180-855 and all data is considered unclassified. Execution of the program and access to the input files is limited to only those users having knowledge about the program, the potential input files, and the user's catalog in which the files reside.

12.3.2.2. Input Formats.

12.3.2.2.1. Length.

The name of the input file may be a maximum of 14 characters in length and is not enclosed in any quote marks.

12.3.2.2.2. Format.

The input file is a multifile file, and all data files within the multifile are of the same format. Each of the data files can logically be divided into three sections: the identifier block, the modification records, and the data records. Refer to section 12.3.3.1 for a detailed description of a data file.

12.3.2.2.3. Labels.

The parameter values can be specified positionally or by parameter name, and must be separated by a space or a comma. Refer to section 12.3.1.3 for a description of the labels for the input parameters.

12.3.2.2.4. Sequence.

When the parameter values are specified without parameter names, the values must be entered in the appropriate sequence described in section 12.3.1.3. When parameter names are used with the parameter values, the order of specification on the procedure call statement is meaningless.

12.3.3. Sample Inputs.

12.3.3.1. The Input File.

The input multifile must be of a specific format. This format is the natural result of reading the digitized data from tape. The format is briefly described here.

12.3.3.1.1. The Identifier Block.

The first 100-words (10 80-character lines) of a data file is the identifier (ID) block. It gives vital information regarding the data records in the file and to what event they pertain.

12.3.3.1.2. The Modification Records.

The modification (MOD) records are written to a data file by the data processing programs as the file proceeds through the data processing system. The MOD records contain information specific to the various processes performed on the data. A value in one of the words of the ID block indicates how many MOD records have been written to the data file. A MOD record, however, consists of three 80-character records. The value from the ID block must be multiplied by three to determine the correct number of MOD records.

12.3.3.1.3. The Data Records.

The data records contain the data in integer format. There are 16 data values per line, or record, of the data file. A value in one of the words of the id block indicates how many data values are in the data file. This value is divided by 16 to determine the number of complete records of data as well as the number of data values in the last record of the file.

12.3.3.2. The Procedure Call Statement.

The following are examples of initiating the "SPECTRA" program.

```
RUN_SPECTRA data_file NST=-20 DN=tk4
```

```
RUN_plot data_file
```

```
Run_plot data_file sst=-5 set=55
```

12.3.4. Output.

12.3.4.1. Use.

The plot output is retained and stored according to event name by Division 7111.

"SPECTRA"
USER MANUAL

12.3.4.2. Destination.

12.3.4.2.1. Line Printer.

There is only one line printer available for receiving output directly from the CYBER 180-855. It is located in the Central Computing Facility in Sandia's primary technical area, Area I, in building 800. The procedure that executes this program directs all output to this line printer.

12.3.4.2.2. Plotter.

The output generated from the "SPECTRA" program is generally disposed to a Versatec plotter. There are several plotters available for use. One is located in the Central Computing Facility in Sandia's primary technical area, Area I, in building 800. Another, and the one to which all plots produced by this program are disposed, is located in Area I, building 806, room 166.

12.3.4.2.3. Terminal.

This program can also dispose the plot output to the terminal running the program, providing the terminal is capable of producing graphics and is supported as a graphics device. This is achieved by specifying a parameter in the procedure that executes the program.

12.3.12.3. Medium.

The plots are stored in hard-copy format, as generated by a Versatec plotter.

12.3.4.4. Disposition.

The plots generated by the "SPECTRA" program are stored by Division 7111 personnel in Area 1, building 806.

12.3.4.5 Sample Outputs.

Figures 12.1, 12.2, and 12.3 represent sample outputs for a station generated by the "SPECTRA" program.

Figure 12.1

AMP X 830.00
TRACK 10
70.0 KHZ
1987-03-13
11:00:00

DEL: .002
MIN: -.3159E-01
MAX: .2976E-01
AZM: 194.6659
RNG: 55362.46
VSN:

NOISE START : -60.0000
NOISE END : -.6080
SIGNAL START : 0.0000
SIGNAL END : 88.0640
BLOCK SIZE : 2048
FREQUENCY INTERVAL : .24414

Figure 12.2

LABOUARK W-30 TOP A-270.

AMP X 830.00
TRACK 10
52.5 KHZ
1987-03-13
11:00:44

DEL: .002
MIN: -.3037E-01
MAX: .3140E-01
AZM: 194.6659
RNG: 55362.46
VSN:

NOISE START : -60.0000
NOISE END : -.6080
SIGNAL START : 0.0000
SIGNAL END : 88.0640
BLOCK SIZE : 2048
FREQUENCY INTERVAL : .24414

Figure 12.3

AMP X 830.00	DEL: .002	NOISE START : -60.0000
TRACK 10	MIN: -.2993E-01	NOISE END : -.6080
40.0 KHZ	MAX: .2698E-01	SIGNAL START : 0.0000
1987-03-13	AZM: 194.6659	SIGNAL END : 88.0640
11:01:25	RNG: 55362.46 -	BLOCK SIZE : 2048
	VSN:	FREQUENCY INTERVAL : .24414

12.3.4.6. Program Termination.

The program terminates normally when an end-of-information is encountered rather than an identifier block for a data file. When the "SPECTRA" program terminates normally, and the plots are to be disposed to the plotter, a message indicating the number of plots generated is displayed on the terminal. The command that transmits the plot file to the NOS operating system for disposition to the plotter is also displayed with all of its associated subcommands. When the terminal is specified as the plot device in the "RUN_SPECTRA" procedure, each plot is generated on the terminal.

12.3.5. Error and Recovery.

There are no special restart procedures. The procedure ensures its own successful termination. This allows multiple restarts of the procedure, regardless of program termination status.

If, while reading the input data file, an end-of-information is encountered anywhere other than at the beginning of the assumed next data file, the program terminates abnormally. The program and the procedure both issue error messages that are displayed on the screen, and the procedure is terminated.

"FILTER"
USER MANUAL

CHAPTER 13

"FILTER"
USER MANUAL

"FILTER"
USER MANUAL

13.1. GENERAL INFORMATION

13.1.1. Summary.

The "FILTER" program builds and applies highpass and lowpass FIR filters to raw channel data as well as to the first and second integrals to produce acceleration, velocity, and displacement output files and plots.

13.1.2. Environment.

This project is sponsored by the NNWSI project. Its development is the responsibility of the Ground Motion and Seismic Division (7111) of the Field Sciences Department (7110). Members of the project within Division 7111 are the sole users of the "FILTER" program.

The "FILTER" program is implemented on Sandia National Laboratories' CDC CYBER 180-855 model computer, located in the Central Computing Facility in Sandia's primary technical area, Area I, in building 800. The CYBER 180-855 operates in a dual state, which allows two operating systems to run concurrently. The "FILTER" program runs on the NOS/VE side of the CYBER 180-855. The current operating system version is 1.1.2 level 649.

13.1.3. References.

13.1.3.1. National Bureau of Standards, "Guidelines for Documentation of Computer Programs and Automated Data Systems," Federal Information Processing Standards (FIPS) Publication #38, February, 1976.

13.1.3.2. Samuel D. Stearns, "Digital Signal Analysis," Rochelle Park, New Jersey: Hayden Book Company, Inc., 1975.

13.1.3.3. Samuel D. Stearns, "Signal Processing Analysis Algorithms" software library, November, 1985.

13.1.3.4. Integrated Software Systems Corporation, "Display Integrated Software System and Plotting LAnguage" software library, current version 9.2, 1505 Sorrento Valley Rd. San Diego, California 92121.

13.2. APPLICATION

13.2.1. Description.

The "FILTER" program reads an input multiframe of raw channel data files and applies the highpass and lowpass FIR filters to each channel file, writing the filtered data to an output file and generating an acceleration plot of the channel file. Each channel file is integrated twice to produce the velocity and displacement data files. These data files are also filtered, written to the output file, and plotted.

13.2.1.1. When to Use this Program.

The "FILTER" program should be used after the highpass and lowpass filter cutoff frequencies have been determined by the "SPECTRA" program to generate plots of the filtered data.

13.2.1.2. Program Functions.

13.2.1.2.1. Parameter Specification.

The user may provide several parameters that constrain the design of the filters and the portions of the raw data to be processed. Any, all, or none of the following parameters can be specified when the filtering program is executed:

1. Highpass frequency.
2. Lowpass frequency.
3. Input delta time.
4. Output delta time.
5. Start time.
6. End time.
7. Arrival time.

If none of the parameters is specified, the preset default values will be used when the program is executed.

13.2.1.2.2. Plot File.

A plot file is generated containing plots of each of the acceleration channel files and the respective velocity and displacement data sets after filtering.

13.2.1.2.3. Output File.

An output multifile is generated containing all filtered data for which a plot is produced. The output file is of a specific format:

- Station x Vertical Gauge Acceleration
- Station x Vertical Gauge Velocity
- Station x Vertical Gauge Displacement
- Station x Radial Gauge Acceleration
- Station x Radial Gauge Velocity
- Station x Radial Gauge Displacement
- Station x Tangential Gauge Acceleration
- Station x Tangential Gauge Velocity
- Station x Tangential Gauge Displacement

The data files will appear in this order for each channel of data in the input multifile that is filtered and plotted.

13.2.2. Operation.

Only those members of the Ground Motion and Seismic Division (7111) who are working on the project and have need to execute the "FILTER" program are authorized to access the procedure that directs the program execution. At the present time, there are three such users.

13.2.3. Equipment.

The "FILTER" program resides on the NOS/VE side of the dual state CDC CYBER 180-855. The CYBER 180-855, under NOS/VE, is configured with 6.9 million words (55.2 million bytes) of memory, and three 895 disks, each with 4 spindles, for a total of 3.6 giga-bytes of online disk space.

13.2.4. Structure.

Software for the "FILTER" program consists of FORTRAN 5 source code with calls to plot routines in a software library called "DISSPLA", both of which are resident on and executed under the control of CDC's NOS/VE.

The "FILTER" program is written in a modular fashion. Each routine performs one basic function or set of related functions, thereby providing a program that is easy to read and modify. The main program interprets the user's selection and directs the appropriate routine to verify and process the input data. When the user's input has been dealt with appropriately, the main program determines the functions to be performed and initiates the proper routine for processing.

"FILTER" USER MANUAL

13.2.5. Performance.

Currently, the filtering program exists and is run interactively under control of the NOS/VE on the dual state CYBER 180-855 computer. The program can be run in batch mode if desired. Program execution is invoked from a procedure. The procedure can easily be modified to start the program in batch mode rather than in interactive mode. No changes need be made to the program to run in batch mode.

13.2.6. Inputs, Processing, and Outputs.

13.2.6.1. Inputs.

Eleven parameters can be specified for the "FILTER" procedure: the input file name, the output file name, the volume serial number of the tape on which the output data file will eventually be written, the highpass frequency, the lowpass frequency, the input delta time, the output delta time, the data start time, the data end time, the arrival time, and the name of device to which the plot file is to be disposed. The name of the input data file is the only required parameter, however.

13.2.6.1.1. Description.

13.2.6.1.1.1. Input File Name.

The input file must be available as a disk file in the NOS/VE system. The input file name must be specified on the procedure call statement.

13.2.6.1.1.2. Output File Name.

The output file is generated during program execution. The output file name may be specified on the procedure call statement. If this parameter is not specified on the procedure call statement, the program uses the default file name SAM_DATA.

13.2.6.1.1.3. Volume Serial Number of the Output Tape.

The output file will eventually be automatically written to tape. The volume serial number of the tape to write does not have to be specified until the tape-writing process has been included as a function of the "RUN SAM" procedure. If this parameter is not specified on the procedure call statement, the program uses a dummy vsn.

13.2.6.1.1.4. Highpass Filter Frequency.

The highpass filter frequency is used in designing the highpass filter. When the highpass frequency is specified on the procedure call statement, it must be given in Hertz, as in ".5". When this parameter is not specified on the procedure call statement, the program will use the default value of .3 Hertz.

13.2.6.1.1.5. Lowpass Filter Frequency.

The lowpass filter frequency is used in designing the lowpass filter. When the lowpass frequency is specified on the procedure call statement, it must be given in Hertz, as in "15". When this parameter is not specified on the procedure call statement, the program will use the default value of 30 Hertz.

13.2.6.1.1.6. Input Delta Time.

When the input delta time is specified on the procedure call statement, it must be given in milli-seconds, as in ".001". When this parameter is not specified on the procedure call statement, the program will use the default value of .002 seconds.

13.2.6.1.1.7. Output Delta Time.

When the output delta time is specified on the procedure call statement, it must be given in milli-seconds, as in ".006". When this parameter is not specified on the procedure call statement, the program will use the default value of .005 seconds.

13.2.6.1.1.8. Data Start Time.

When the data start time is specified on the procedure call statement, it must be given in seconds; half-seconds can be specified, as in "-10.5". When this parameter is not specified on the procedure call statement, the program will use the default value of -20. seconds.

13.2.6.1.1.9. Data End Time.

When the data end time is specified on the procedure call statement, it must be given in seconds; half-seconds can be specified, as in "85.5". When this parameter is not specified on the procedure call statement, the program will use the default value of 100. seconds.

13.2.6.1.1.10. Data Arrival Time.

When the data arrival time is specified on the procedure call statement, it must be given in seconds; half-seconds can be specified, as in "5.5". When this parameter is not specified on the procedure call statement, the program will use the default value of 0. seconds.

13.2.6.1.1.11. Plot Device.

When this parameter is specified on the procedure call statement, it must be a three-character code recognized by the NOS/VE system as a graphics output device. When this parameter is not specified, it is set to "HC1," which disposes the plot output to the Versatec plotter.

13.2.6.2. Processing.

13.2.6.2.1. Read the Parameter List.

The program reads the input parameter values from the parameter file created by the "RUN SAM" procedure from the parameter values specified on the procedure call statement. When a value for an optional parameter is not specified in the procedure call statement, the program uses the default value for the parameter.

13.2.6.2.2. Preparation for Filtering the Data.

13.2.6.2.2.1. Building the Filters.

FIR filters are designed for both the lowpass and highpass filters. Bandwidth and frequency cutoff are derived using given constraints, and the filter coefficients are determined. The methods employed are taken from algorithms designed and described in "Digital Signal Analysis", by Samuel D. Stearns, 1975, and from routines written by Samuel D. Stearns, available in a software library called "Signal Processing Analysis Algorithms."

13.2.6.2.2.2. Getting the Data File.

The input data file is read from the beginning-of-file position. An identifier (id) block is read from the data file and the following values are extracted: the number of data values in the file, the number of modification records in the file, the delta time, the data start time, the data stop time, the minimum data value, the maximum data value, and the multiplier to reconstruct the data. The modification records are read and skipped. The data is then read from the data file. The delta time from the id block is compared to the input data delta time specified in the input parameters. An error message is issued if the times differ,

and no data from the data set is processed; execution continues with the reading of the next data set in the input multifile.

If the data set is from a velocity gauge rather than from an acceleration gauge, the coefficients of a hanned differentiator are calculated for later use. The data set is expanded from its integer format to real format, using the conversion constant read from the id block. Using the start time from the id block and the start and end times specified in the input parameter list or by their default values, the indices for the start and end times for the data segment are calculated.

The data segment to be filtered is then extracted from the data set.

13.2.6.2.3. Filtering the Acceleration Data.

The data set is applied to the lowpass filter and then resampled at the the output data delta time. The resulting data set is applied to the highpass filter. Then the data is adjusted by the simple average of the data between the start and arrival times. The data is written to the output file and the plot file.

13.2.6.2.4. Filtering the Velocity Data.

The data set containing the filtered acceleration data is integrated to produce a velocity data set. The velocity data is adjusted by the simple average of the data between the start and arrival times. A trend defined by the the data between the start and end times is then determined and removed from the data set. The data is written to the output file and the plot file.

13.2.6.2.5. Filtering the Displacement Data.

The data set containing the filtered velocity data is integrated to produce a displacement data set. A parabola defined by the data between the arrival and end times is determined and removed from the data set. The displaced data is then adjusted by the simple average of the data between the arrival and end times. The data is applied to the highpass filter, and again adjusted by the simple average of the data between the arrival and end times. The resulting data is written to the output file and the plot file.

13.2.6.3. Outputs.

The "FILTER" program creates a plot file containing the acceleration, velocity, and displacement plots of each of the data files within the input multifile. An output file containing the filtered data for the

"FILTER"
USER MANUAL

acceleration, velocity, and displacement data sets and their associated identifier blocks and modification records is also created. The output multifile is of the format described in section 13.2.2.3.

13.3. PROCEDURES AND REQUIREMENTS

13.3.1. Initiation.

13.3.1.1. The Procedure.

The "FILTER" program is initiated and executed under the control of a procedure named "RUN_SAM." "RUN_SAM" performs all of the commands necessary to execute the "FILTER" program and dispose the plot file to the specified output device, which simplifies the user's task of running the program. The parameters that comprise the parameter input file for the program are specified on the procedure call statement. With these given values, the procedure generates the parameter input file that is used by the program, and also issues the command to dispose the plot file to the specified device.

13.3.1.2. Getting a Copy of the Procedure.

Both "RUN_SAM" and the program source "FILTER" are resident in disk space allocated to a particular member of the Ground Motion and Seismic Division (7111). For a user other than the owner to access the procedure, the user name of the owner must be specified on the command that retrieves the file. The following describes the steps necessary for a user to get a copy of the procedure:

```
COPY_FILE .MLSANDE.COMMAND_CATALOG.RUN_SAM RUN_SAM
```

The "COPY_FILE" command will work properly only if the procedure file has been made available for access by the user requesting it. This command will make a copy of the procedure in the user's catalog and will name it "RUN_SAM." When the procedure is executed, it will access all of the files necessary to execute the program, including the program source.

13.3.1.3. Invoking the Procedure.

"RUN_SAM" creates a parameter input file for the program from the parameter list specified on the statement that invokes the procedure. The "RUN_SAM" procedure will accept the eleven parameters described in section 13.2.6.1.1. The parameters can be specified positionally or by parameter name, and must be separated by a space or a comma. Parameters specified by a name have the following form:

```
name=value
```

where name is the parameter name and value is the parameter value. This is the command that invokes the procedure:

"FILTER"
USER MANUAL

```
RUN SAM DF=data_file OF=sam data VSN=volume serial_number...  
      HP=high_pass_freq LP=low pass_freq IDT=input delta time...  
      ODT=output_delta_time DST=start_time DET=end_time...  
      AT=arrival_time DN=device_name
```

13.3.1.4. Compilation and Execution.

The procedure compiles the program source. When the source does not compile normally, a listing of the errors is displayed on the screen and the procedure is terminated. When the source compiles normally, program execution is initiated.

13.3.2. Input.

13.3.2.1. Procedure Parameters.

The "FILTER" procedure will accept the eleven parameters described in section 13.2.6.1.1

The procedure creates a parameter input file for the program from the parameter values specified on the procedure call statement. The name of the input data file is the only required parameter. The default value for a parameter is used if an optional parameter is not entered. Refer to section 13.2.6.1. for a description of the parameters and their default values. The parameter names need not be included when the parameter values are given positionally as shown above in section 13.3.1.3.

13.3.2.2. Requirements.

13.3.2.2.1. Frequency.

The "PLOT" program should be run to generate plots of the raw data prior to any processing. "PLOT" is also run during processing to generate expanded plots. The appropriate procedures that initiate the processing programs also initiate the execution of the "PLOT" program.

13.3.2.2.2. Origin.

The input multifile is generated in several ways. The data can be created by the program that reads the data from the tape after digitization, or it can be the resulting data file after executing one of the processing programs. The program parameters are selected by the user when the procedure is initiated.

13.3.2.2.3. Medium.

The input data file is available to the program as it exists in a file on disk. The program parameters are specified on the command line entered on the terminal.

13.3.2.2.4. Restrictions.

There are no security restrictions placed on the operation of the "PLOT" program, as it resides in the open-partition CYBER 180-855 and all data is considered unclassified. Execution of the program and access to the input files is limited to only those users having knowledge about the program, the potential input files, and the user's catalog in which the files reside.

13.3.2.3. Input Formats.

13.3.2.3.1. Length.

The name of the input file may be a maximum of 14 characters in length and is not enclosed in any quote marks. The name of the output file may be a maximum of 14 characters in length and is not enclosed in any quote marks.

13.3.2.3.2. Format.

The input file is a multifile file, and all data files within the multifile are of the same format. The output file is also a multifile file, and all data files within the multifile are of the same format as the input multifile. Each of the data files can logically be divided into three sections: the identifier block, the modification records, and the data records. Refer to section 13.3.3.1 for a detailed description of a data file.

13.3.2.3.3. Labels.

The parameter values can be specified positionally or by parameter name, and must be separated by a space or a comma. Refer to section 13.3.1.3 for a description of the labels for the input parameters.

13.3.2.3.4. Sequence.

When the parameter values are specified without parameter names, the values must be entered in the appropriate sequence described in section 13.3.1.3. When parameter names are used with the parameter values, the order of specification on the procedure call statement is meaningless.

13.3.3. Sample Inputs.

13.3.3.1. The Input File.

The input multifile must be of a specific format. This format is the natural result of reading the digitized data from tape, as well as from executing the "SPECTRA" program. The format is briefly described here.

13.3.3.1.1. The Identifier Block.

The first 100-words (10 80-character lines) of a data file is the identifier (ID) block. It gives vital information regarding the data records in the file and to what event they pertain.

13.3.3.1.2. The Modification Records.

The modification (MOD) records are written to a data file by the data processing programs as the file proceeds through the data processing system. The MOD records contain information specific to the various processes performed on the data. A value in one of the words of the ID block indicates how many mod records have been written to the data file. A MOD record, however, consists of three 80-character records. The value from the id block must be multiplied by three to determine the correct number of MOD records.

13.3.3.1.3. The Data Records.

The data records contain the data in integer format. There are 16 data values per line, or record, of the data file. A value in one of the words of the id block indicates how many data values are in the data file. This value is divided by 16 to determine the number of complete records of data as well as the number of data values in the last record of the file.

13.3.3.2. The Procedure Call Statement.

The following are examples of initiating the "FILTER" program.

```
RUN_FILTER data_file out_file idt=.001
```

```
RUN_plot data_file vsn=12314
```

```
Run_plot DF=data_file AT=-5. DN=tk4
```

"FILTER"
USER MANUAL

13.3.4. Output.

13.3.4.1. Use.

The hard-copies of the plots are retained and stored according to event name by Division 7111. The output file containing the data necessary to re-generate the filtered data plots is also stored in a disk file belonging to a member of Division 7111. The output file is eventually written to magnetic tape.

13.3.4.2. Destination.

13.3.4.2.1. Line Printer.

There is only one line printer available for receiving output directly from the CYBER 180-855. It is located in the Central Computing Facility in Sandia's primary technical area, Area I, in building 800. The procedure that executes this program directs all output to this line printer.

13.3.4.2.2. Plotter.

The output generated from the "FILTER" program is generally disposed to a Versatec plotter. There are several plotters available for use. One is located in the Central Computing Facility in Sandia's primary technical area, Area I, in building 800. Another, and the one to which all plots produced by this program are disposed, is located in Area I, building 806, room 166.

13.3.4.2.3. Terminal.

This program can also dispose the plot output to the terminal running the program, providing the terminal is capable of producing graphics and is supported as a graphics device. This is achieved by specifying a parameter in the procedure that executes the program.

13.3.4.3. Medium.

The plots are stored in hard-copy format, as generated by a Versatec plotter. The output file containing the data necessary to re-generate the filtered data plots is stored in the file space of a member of Division 7111 until it is written to magnetic tape. After the data is written to tape, the disk file is purged.

13.3.4.4. Disposition.

The plots generated by the "FILTER" program are stored by Division 7111 personnel in Area 1, building 806. The magnetic tapes containing the filtered data are stored by Division 7111.

13.3.4.5 Sample Outputs.

13.3.4.5.1. Plot.

Figures 13.1 through 13.9 represent sample outputs for a station generated by the "FILTER" program.

13.3.4.5.2. Output Data File.

Figure 13.10 represents the format of the output file from the "FILTER" program.

Figure 13.1

AMP X 830.00
TRACK 10
70.0 KHZ
1987-03-13
10:12:39

DEL: J005
MIN: -.3148E-01
MAX: .2937E-01
AZM: 194.6659
RNG: 55362.46
VSN: 000001

LPF: 27.
HPF: .300
BWL: 6.0
BWH: .200
HLL: 167
HLH: 1999

Figure 13.2

AMP X 830.00
TRACK 10
70.0 KHZ
1987-03-13
10:13:06

DEL: .005
MIN: -.5740E-02
MAX: .4471E-02
AZM: 194.6659
RNG: 55362.46
VSN: 000001

LPF: 27.
HPF: .300
BWL: 6.0
BWH: .200
HLL: 167
HLH: 1999

Figure 13.3

AMP X 830.00
TRACK 10
70.0 KHZ
1987-03-13
10:13:48

DEL: .005
MIN: -.1746E+00
MAX: .1888E+00
AZM: 194.6659
RNG: 55362.46
VSN: 000001

LPF: 27.
HPF: .300
BWL: 6.0
BWH: .200
HLL: 167
HLH: 1999

Figure 13.4

Figure 13.5

LABQUARK W-30 TOP A-270. VELOCITY

AMP X 830.00
IRACK 10
52.5 KHZ
1987-03-13
10:15:33

DEL: .005
MIN: -.9055E-02
MAX: .7741E-02
AZM: 194.6659
RNG: 55362.46
VSN: 000001

LPF: 27.
HPF: .300
BWL: 6.0
BWH: .200
HLL: 167
HLH: 1999

Figure 13.6

Figure 13.7

Figure 13.8

Figure 13.9

"FILTER"
USER MANUAL

Figure 13.10

1

SCAN OF TAPE scan_data

FILE NO.	DATA ID	STATION	ORIENT	EVENT	ROTATE	APP	TR FREQ	START	END
1	FILTER 2	W-30 TOP	AV	LABQUARK		830.00	TR1070.0	-20.0000	99.9980
2	INTEG 2	W-30 TOP	AV	LABQUARK		830.00	TR1070.0	-20.0000	99.9980
3	INTEG2 2	W-30 TOP	AV	LABQUARK		830.00	TR1070.0	-20.0000	99.9980
4	FILTER 2	W-30 TOP	A-270.	LABQUARK		830.00	TR1052.5	-20.0000	99.9980
5	INTEG 2	W-30 TOP	A-270.	LABQUARK		830.00	TR1052.5	-20.0000	99.9980
6	INTEG2 2	W-30 TOP	A-270.	LABQUARK		830.00	TR1052.5	-20.0000	99.9980
7	FILTER 2	W-30 TOP	A-0.	LABQUARK		830.00	TR1040.0	-20.0000	99.9980
8	INTEG 2	W-30 TOP	A-0.	LABQUARK		830.00	TR1040.0	-20.0000	99.9980
9	INTEG2 2	W-30 TOP	A-0.	LABQUARK		830.00	TR1040.0	-20.0000	99.9980

ATA WAS PROCESSED IN ACCORDANCE WITH ETP-7111-1.

SCAN - NORMAL TERMINATION.
9 FILES PROCESSED.

13.3.4.6. At Program Termination.

The program terminates normally when an end-of-information is encountered rather than an identifier block for a data file. When the "FILTER" program terminates normally, and the plots are to be disposed to the plotter, a message indicating the number of plots generated is displayed on the terminal. The command that transmits the plot file to the NOS operating system for disposition to the plotter is also displayed with all of its associated subcommands. When the terminal is specified as the plot device in the "RUN_SAM" procedure, each plot is generated on the terminal.

13.3.5. Error and Recovery.

There are no special restart procedures. The procedure ensures its own successful termination. This allows multiple restarts of the procedure, regardless of program termination status.

If, while reading the input data file, an end-of-information is encountered anywhere other than at the beginning of the assumed next data file, the program terminates abnormally. The program and the procedure both issue error messages that are displayed on the screen, and the procedure terminates.

CHAPTER 14

"ROTATE"
USER MANUAL

"ROTATE" USER MANUAL

14.1. GENERAL INFORMATION

14.1.1. Summary.

The "ROTATE" program performs a transformation of channel data that separates radial and tangential signal components. This transform is logically equivalent to rotating a station's radial gauge to point directly to the signal source, and its tangential gauge to a true tangent to the source.

14.1.2. Environment.

This project is sponsored by the NNWSI project. Its development is the responsibility of the Ground Motion and Seismic Division (7111) of the Field Sciences Department (7110). Members of the project within Division 7111 are the sole users of the "FILTER" program.

The "ROTATE" program is implemented on Sandia National Laboratories' CDC CYBER 180-855 model computer, located in the Central Computing Facility in Sandia's primary technical area, Area I, in building 800. The CYBER 180-855 operates in a dual state, which allows two operating systems to run concurrently. The "ROTATE" program runs on the NOS/VE side of the CYBER 180-855. The current operating system version is 1.1.2 level 649.

14.1.3. References.

14.1.3.1. National Bureau of Standards, "Guidelines for Documentation of Computer Programs and Automated Data Systems," Federal Information Processing Standards (FIPS) Publication #38, February, 1976.

14.1.3.2. Integrated Software Systems Corporation, "Display Integrated Software System and Plotting LAnguage" software library, current version 9.2, 1505 Sorrento Valley Rd. San Diego, California 92121.

14.2. APPLICATION

14.2.1. Description.

The "ROTATE" program performs a transformation of channel data that separates radial and tangential signal components. This transform is logically equivalent to rotating a station's radial gauge to point directly to the signal source, and its tangential gauge to a true tangent to the source.

14.2.1.1. When to Use this Program.

The "ROTATE" program should be used after the filtering program "FILTER" has been run, to generate plots of the rotated radial and tangential data and to create an output file containing the channels' rotated data.

14.2.1.2. Program Functions.

14.2.1.2.1. Plot File.

A plot file is generated containing plots of a channel's acceleration data and the respective velocity and displacement data sets after rotation for each channel of data in the input file.

14.2.1.2.2. Output File.

An output multifile is generated containing the acceleration data and the rotated radial and tangential data for each channel in the input file for which a plot was generated. The output multifile is of the following format:

```
Station x Vertical Gauge Acceleration
Station x Vertical Gauge Rotated Velocity
Station x Vertical Gauge Rotated Displacement
Station x Radial Gauge Acceleration
Station x Radial Gauge Rotated Velocity
Station x Radial Gauge Rotated Displacement
Station x Tangential Gauge Acceleration
Station x Tangential Gauge Rotated Velocity
Station x Tangential Gauge Rotated Displacement
```

The data files will appear in this order for each channel of data in the input multifile that is rotated and plotted.

"ROTATE" USER MANUAL

14.2.2. Operation.

Only those members of the Ground Motion and Seismic Division (7111) who are working on the project and have need to execute the "ROTATE" program are authorized to access the procedure that directs the program execution. At the present time, there are three such users.

14.2.3. Equipment.

The "ROTATE" program resides on the NOS/VE side of the dual state CDC CYBER 180-855. The CYBER 180-855, under NOS/VE, is configured with 6.9 million words (55.2 million bytes) of memory, and three 895 disks, each with 4 spindles, for a total of 3.6 giga-bytes of online disk space.

14.2.4. Structure.

Software for the "ROTATE" program consists of FORTRAN 5 source code with calls to plot routines in a software library called "DISSPLA", both of which are resident on and executed under the control of CDC's NOS/VE.

The "ROTATE" program is written in a modular fashion. Each routine performs one basic function or set of related functions, thereby providing a program that is easy to read and modify. The main program interprets the user's selection and directs the appropriate routine to verify and process the input data. When the user's input has been dealt with appropriately, the main program determines the functions to be performed and initiates the proper routine for processing.

14.2.5. Performance.

Currently, the rotation program exists and is run interactively under control of the NOS/VE on the dual state CYBER 180-855 computer. The program can be run in batch mode if desired. Program execution is invoked from a procedure. The procedure can easily be modified to start the program in batch mode rather than in interactive mode. No changes need be made to the program to run in batch mode.

14.2.6. Inputs, Processing, and Outputs.

14.2.6.1. Inputs.

Five parameters can be specified for the "ROTATE" procedure: the input file name, the output file name, the volume serial number of the tape on which the output data file will eventually be written, the number of the event for which data is being processed, and the name of the device to which the plot file is to be disposed. The name of the input data file is the only required parameter, however.

14.2.6.1.1. Description.

14.2.6.1.1.1. Input File Name.

The input file must be available as a disk file in the NOS/VE system. The input file name must be specified on the procedure call statement.

14.2.6.1.1.2. Output File Name.

The output file is generated during program execution. The output file name may be specified on the procedure call statement. If this parameter is not specified on the procedure call statement, the program uses the default file name "ROT_DATA."

14.2.6.1.1.3. Volume Serial Number of the Output Tape.

The output file will eventually be automatically written to tape. The volume serial number of the tape to write does not have to be specified until the tape-writing process has been included as a function of the "RUN_SAM" procedure. If this parameter is not specified on the procedure call statement, the program uses a vsn value of "000001."

14.2.6.1.1.4. The Event Number.

The event number is entered as a three-digit number; the procedure converts it to the appropriate format for the program: "NxxxE," where xxx is the event number. This parameter is required, and must be specified on the procedure call statement for the procedure to execute.

14.2.6.1.1.5. Plot Device.

When this parameter is specified on the procedure call statement, it must be a three-character code recognized by the NOS/VE system as a graphics output device. When this parameter is not specified, it is set to "HC1", which disposes the plot output to the Versatec plotter.

14.2.6.2. Processing.

14.2.6.2.1. Read the Parameter List.

The program reads the input parameter values from the parameter file created by the "RUN_ROTATE" procedure from the parameter values specified on the procedure call statement. When a value for

an optional parameter is not specified in the procedure call statement, the program uses the default value for the parameter.

14.2.6.2.2. Initialize the Data Base.

The WTSI data base files for the event that were migrated by the secondary procedure described in section 14.3.3.1.2. are initialized for read-only access. The event record is then read from the data base.

14.2.6.2.3. Preparation for Rotating the Data.

14.2.6.2.3.1. Get the Data File.

The input data file is read from the beginning-of-file position. An identifier (id) block is read from the data file and the number of data values and modification records in the file are read from the id block. The modification records and the data records are then read from the data file. The id block, the modification records, and the data records are written to the output file.

14.2.6.2.3.2. Verify the Data File.

The event name in the ID block of the input data file is compared to the event name in the event record from the data base. If there is a discrepancy, the program aborts and issues an error message.

14.2.6.2.3.3. Read Data Base Records.

The data base event record file is read until the record for the station corresponding to the station in the data file is found. Then the data base station record file is read until the appropriate station record for the date of the event is found. If the appropriate records from the files are not found, the program aborts and issues appropriate error messages.

14.2.6.2.3.4. Read Station Orientations.

When the station record for the event is found, the orientations of the radial and tangential gauges are read from the record. The station orientation is necessary to the calculation of the rotation angles to be applied to the data.

14.2.6.2.3.5. Write Data to the Output File.

The input data file is written to a scratch file determined by the data in the file: the vertical, radial, and tangential gauge data files are each written to different output files.

14.2.6.2.3.6. Read All Files for the Station.

All remaining data files for the station are read from the input file. Each file is written to the output file corresponding to the gauge type of the data, as described in section 14.4.3.2.2.5. If, while reading the remaining data files for the station from the input file, a file containing a different station name is read, the program proceeds to the next step, as this indicates that all data files for the station have been extracted from the input file. This will occur only when data for the radial or tangential gauge is missing from the input file.

14.2.6.2.3.7. The Output Files.

When data for all gauges are present, there are nine data files written to the three output files. It is only in this case that a rotation of the radial and tangential gauges can be performed. In some cases, the data for either the radial or the tangential gauge is missing from the input data file. In this instance, the appropriate output file will contain no data files, and thus the rotation can not be performed.

14.2.6.2.4. Rotate the Data.

The angles of rotation to be applied to the radial and tangential gauges are determined based on the event and the station coordinates. Data for the vertical gauge is not rotated. Data for the radial gauge is rotated to point directly to the signal source while data for the tangential gauge is rotated to a true tangent to the signal source. Each data file for each of the radial and tangential gauges is rotated using the determined angles of rotation.

14.2.6.3. Outputs.

The "ROTATE" program creates a plot file containing a station's vertical gauge data, and the rotated radial and tangential gauge data for each of the stations within the input multifile. An output file containing the vertical data and the rotated radial and tangential data files with their associated identifier blocks is also created. The format of this output file is described in section 14.2.2.2.

14.3. PROCEDURES AND REQUIREMENTS

14.3.1. Initiation.

14.3.1.1. The Procedure.

The "ROTATE" program is initiated and executed under the control of a procedure named "RUN_ROTATE." "RUN_ROTATE" performs all of the commands necessary to execute the "ROTATE" program and dispose the plot file to the specified output device, which simplifies the user's task of running the program. A secondary procedure is called from the "RUN_ROTATE" procedure; this is not user invoked. The parameters the comprise the parameter input file for the program are specified on the procedure call statement. With these given values, the procedure generates the parameter input file that is used by the program, and also issues the command to dispose the plot file to the specified device.

14.3.1.2. Getting a Copy of the Procedure.

Both "RUN_ROTATE" and the program source "ROTATE" are resident in disk space allocated to a particular member of the Ground Motion and Seismic Division (7111). This user's disk space is called a "catalog" in the NOS/VE operating system. For a user other than the owner to access the procedure, the user name of the owner must be specified on the command that retrieves the file. The following describes the steps necessary for a user to get a copy of the procedure:

```
COPY_FILE .MLSANDE.COMMAND_CATALOG.RUN_ROTATE RUN_ROTATE
```

The "COPY_FILE" command will work properly only if the procedure file has been made available for access by the user requesting it. Refer to the "SCL for NOS/VE Quick Reference" manual for further information regarding copying files from another user's catalog. This command will make a copy of the procedure in the user's catalog and will name it "RUN_ROTATE." When the procedure is executed, it accesses all of the files necessary to execute the program.

14.3.1.3. Invoking the Procedure.

"RUN_ROTATE" creates a parameter input file for the program from the parameter list specified on the statement that invokes the procedure. The "RUN_ROTATE" procedure will accept the five parameters described in section 14.2.6.1.1. The parameters can be specified positionally or by parameter name, and must be separated by a space or a comma. Parameters specified by a name have the following form:

name=value

"ROTATE" USER MANUAL

where name is the parameter name and value is the parameter value.
This is the command that invokes the procedure:

```
RUN_ROTATE DF=data_file OF=rot_data VSN=volume_serial_number...  
           EN=event_number DN=device_name
```

14.3.1.4. Compilation and Execution.

The procedure compiles the program source. When the source does not compile normally, a listing of the errors is displayed on the screen and the procedure is terminated. When the source compiles normally, program execution is initiated.

14.3.2. Input.

14.3.2.1. Procedure Parameters.

The "ROTATE" procedure will accept five parameters: the input file name, the output file name, the volume serial number of the tape on which the output data file will eventually be written, the event number, and the name of the plot device to receive the plots.

The procedure creates a parameter input file for the program from the parameter values specified on the procedure call statement. The name of the input data file is the only required parameter. The default value for a parameter is used if an optional parameter is not entered. Refer to section 14.2.6.1. for a description of the parameters and their default values. The parameter names need not be included when the parameter values are given positionally as shown above in section 14.3.1.3.

14.3.2.2. Requirements.

14.3.2.2.1. Frequency.

The "ROTATE" program should be run after the filtering program "FILTER" has been run, to generate plots of the rotated gauges for each station in the input data file.

14.3.2.2.2. Origin.

The input multifile is generated by the filtering program "FILTER." The data files in the multifile are in the order that the "ROTATE" program expects. The program parameters are selected by the user when the procedure is initiated.

14.3.2.2.3. Medium.

The input data file is available to the program as it exists in a file on disk. The program parameters are specified on the command line entered on the terminal.

14.3.2.2.4. Restrictions.

There are no security restrictions placed on the operation of the "ROTATE" program, as it resides in the open-partition CYBER 180-855 and all data is considered unclassified. Execution of the program and access to the input files is limited to only those users having knowledge about the program, the potential input files, and the user's catalog in which the files reside.

14.3.2.3. Input Formats.

14.3.2.3.1. Length.

The name of the input file may be a maximum of 14 characters in length and is not enclosed in any quote marks. The name of the output file may be a maximum of 14 characters in length and is not enclosed in any quote marks.

14.3.2.3.2. Format.

The input file is a multifile file, and all data files within the multifile are of the same format. The output file is also a multifile file, and all data files within the multifile are of the same format as the input multifile. Each of the data files can logically be divided into three sections: the identifier block, the modification records, and the data records. Refer to section 14.3.3.1 for a detailed description of a data file.

14.3.2.3.3. Labels.

The parameter values can be specified positionally or by parameter name, and must be separated by a space or a comma. Refer to section 14.3.1.3 for a description of the labels for the input parameters.

14.3.2.3.4. Sequence.

When the parameter values are specified without parameter names, the values must be entered in the appropriate sequence described in section 14.3.1.3. When parameter names are used with the parameter values, the order of specification on the procedure call statement is meaningless.

14.3.3. Sample Inputs.

14.3.3.1. The Input File.

The input multifile must be of a specific format. This format is the natural result of running the filtering program "FILTER." The format is briefly described here.

14.3.3.1.1. The Identifier Block.

The first 100-words (10 80-character lines) of a data file is the identifier (ID) block. It gives vital information regarding the data records in the file and to what event they pertain.

14.3.3.1.2. The Modification Records.

The modification (MOD) records are written to a data file by the data processing programs as the file proceeds through the data processing system. The MOD records contain information specific to the various processes performed on the data. A value in one of the words of the ID block indicates how many MOD records have been written to the data file. A MOD record, however, consists of three 80-character records. The value from the ID block must be multiplied by three to determine the correct number of MOD records.

14.3.3.1.3. The Data Records.

The data records contain the data in integer format. There are 16 data values per line, or record, of the data file. A value in one of the words of the ID block indicates how many data values are in the data file. This value is divided by 16 to determine the number of complete records of data as well as the number of data values in the last record of the file.

14.3.3.2. The Procedure Call Statement.

The following are examples of initiating the "ROTATE" program.

```
RUN_ROTATE data_file out_file
```

```
RUN_rotate data_file vsn=12314
```

```
Run_Rotate DF=data_file IDT=.005 AT=5. DN=tk4
```

14.3.4. Output.

14.3.4.1. Use.

The hard-copies of the plots are retained and stored according to event name by Division 7111. The output file containing the data necessary to re-generate the rotated data plots is also stored in a disk file belonging to a member of Division 7111. The output file is eventually written to magnetic tape.

14.3.4.2. Destination.

14.3.4.2.1. Line Printer.

There is only one line printer available for receiving output directly from the CYBER 180-855. It is located in the Central Computing Facility in Sandia's primary technical area, Area I, in building 800. The procedure that executes this program directs all output to this line printer.

14.3.4.2.2. Plotter.

The output generated from the "ROTATE" program is generally disposed to a Versatec plotter. There are several plotters available for use. One is located in the Central Computing Facility in Sandia's primary technical area, Area I, in building 800. Another, and the one to which all plots produced by this program are disposed, is located in Area I, building 806, room 166.

14.3.4.2.3. Terminal.

This program can also dispose the plot output to the terminal running the program, providing the terminal is capable of producing graphics and is supported as a graphics device. This is achieved by specifying a parameter in the procedure that executes the program.

14.3.4.3. Medium.

The plots are stored in hard-copy format, as generated by a Versatec plotter. The output file containing the data necessary to re-generate the filtered data plots is stored in the file space of a member of Division 7111 until it is written to magnetic tape. After the data is written to tape, the disk file is purged.

14.3.4.4. Disposition.

The plots generated by the "ROTATE" program are stored by Division 7111 personnel in Area 1, building 806. The magnetic tapes containing the filtered data are stored by Division 7111.

14.3.4.5 Sample Outputs.

14.3.4.5.1. Plot.

Figures 14.1 through 14.9 represent sample outputs for a station generated by the "ROTATE" program.

14.3.4.5.2. Output Data File.

Figure 14.10 represents the format of the output file from the "ROTATE" program.

Figure 14.1

AMP X 830.00
TRACK 10
70.0 KHZ
1987-03-12
13:57:59

DEL: .005
MIN: -.3148E-01
MAX: .2937E-01
AZM: 194.6659
RNG: 55362.46
VSN: 123456

LPF: 27.
HPF: .300
BWL: 6.0
BWH: .200
HLL: 167
HLH: 1999

Figure 14.2

AMP X 830.00
TRACK 10
70.0 KHZ
1987-03-12
13:58:10

DEL: .005
MIN: -.5740E-02
MAX: .4471E-02
AZM: 194.6659
RNG: 55362.46
VSN: 123456

LPF: 27.
HPF: .300
BWL: 6.0
BWH: .200
HLL: 167
HLH: 1999

Figure 14.3

Figure 14.4

AMP X 830.00
TRACK 10
52.5 KHZ
1987-03-12
13:58:33

DEL:.005
MIN:-.3019E-01
MAX:.3135E-01
AZM:194.6659
RNG: 55362.46
VSN:123456

LPF: 27.
HPF:.300
BWL: 6.0
BWH:.200
HLL: 167
HLH:1999

Figure 14.5

AMP X 830.00
TRACK 10
52.5 KHZ
1987-03-12
13:58:46

DEL: .005
MIN: -.9055E-02
MAX: .7741E-02
AZM: 194.6659
RNG: 55362.46
VSN: 123456

LPF: 27.
HPF: .300
BWL: 6.0
BWH: .200
HLL: 167
HLH: 1999

Figure 14.6

Figure 14.7

AMP X 830.00
TRACK 10
40.0 KHZ
1987-03-12
13:59:12

DEL: .005
MIN: -.2867E-01
MAX: .2724E-01
AZM: 194.6659
RNG: 55362.46
VSN: 123456

LPF: 27.
HPF: .300
BWL: 6.0
BWH: .200
HLL: 167
HLH: 1999

Figure 14.8

AMP X 830.00
TRACK 10
40.0 KHZ
1987-03-12
13:59:25

DEL: .005
MIN: -.4137E-02
MAX: .4599E-02
AZM: 194.6659
RNG: 55362.46
VSN: 123456

LPF: 27.
HPF: .300
BWL: 6.0
BWH: .200
HLL: 167
HLH: 1999

Figure 14.9

AMP X 830.00
TRACK 10
40.0 KHZ
1987-03-12
13:59:38

DEL: .006
MIN: -.1572E+00
MAX: .1501E+00
AZM: 194.6659
RNG: 55362.46
VSN: 123456

LPF: 27.
HPF: .300
BWL: 6.0
BWH: .200
HLL: 167
HLH: 1999

"ROTATE"
USER MANUAL

Figure 14.10

1

SCAN OF TAPE scan_data

FILE NO.	DATA ID	STATION	ORIENT	EVENT	ROTATE	APP	TR FREQ	START	END
1	FILTER 2	W-30 TOP	AV	LABQUARK		830.00	TR1070.0	-20.0000	99.9980
2	INTEG 2	W-30 TOP	AV	LABQUARK		830.00	TR1070.0	-20.0000	99.9980
3	INTEG2 2	W-30 TOP	AV	LABQUARK		830.00	TR1070.0	-20.0000	99.9980
4	ROTATE 3	W-30 TOP	AR	LABQUARK	284 -194	830.00	TR1052.5	-20.0000	99.9980
5	INTEG 3	W-30 TOP	AR	LABQUARK	284 -194	830.00	TR1052.5	-20.0000	99.9980
6	INTEG2 3	W-30 TOP	AR	LABQUARK	284 -194	830.00	TR1052.5	-20.0000	99.9980
7	ROTATE 3	W-30 TOP	AT	LABQUARK	284 -284	830.00	TR1040.0	-20.0000	99.9980
8	INTEG 3	W-30 TOP	AT	LABQUARK	284 -284	830.00	TR1040.0	-20.0000	99.9980
9	INTEG2 3	W-30 TOP	AT	LABQUARK	284 -284	830.00	TR1040.0	-20.0000	99.9980

THIS DATA WAS PROCESSED IN ACCORDANCE WITH ETP-7111-1.

SCAN - NORMAL TERMINATION.
9 FILES PROCESSED.

14.3.4.6. At Program Termination.

The program terminates normally when an end-of-information is encountered rather than an identifier block for a data file. When the "ROTATE" program terminates normally, and the plots are to be disposed to the plotter, a message indicating the number of plots generated is displayed on the terminal. The command that transmits the plot file to the NOS operating system for disposition to the plotter is also displayed with all of its associated subcommands. When the terminal is specified as the plot device in the "RUN_ROTATE" procedure, each plot is generated on the terminal.

14.3.5. Error and Recovery.

There are no special restart procedures. The procedure ensures its own successful termination. This allows multiple restarts of the procedure, regardless of program termination status.

If, while reading the input data file, an end-of-information is encountered anywhere other than at the beginning of the assumed next data file, the program terminates abnormally. The program and the procedure both issue error messages that are displayed on the screen, and the procedure terminates.

"PSRV"
USER MANUAL

CHAPTER 15

"PSRV"

USER MANUAL

15.1. GENERAL INFORMATION

15.1.1. Summary.

The "PSRV" program calculates the pseudo relative response velocity for acceleration data.

15.1.2. Environment.

This project is sponsored by the NNWSI project. Its development is the responsibility of the Ground Motion and Seismic Division (7111) of the Field Sciences Department (7110). Members of the project within Division 7111 are the sole users of the "PSRV" program.

The "PSRV" program is implemented on Sandia National Laboratories' CDC CYBER 180-855 MOD el computer, located in the Central Computing Facility in Sandia's primary technical area, Area I, in building 800. The CYBER 180-855 operates in a dual state, which allows two operating systems to run concurrently. The "PSRV" program runs on the NOS/VE side of the CYBER 180-855. The current operating system version is 1.1.2 level 649.

15.1.3. References.

15.1.3.1. National Bureau of Standards, "Guidelines for Documentation of Computer Programs and Automated Data Systems," Federal Information Processing Standards (FIPS) Publication #38, February, 1976.

15.1.3.2. Integrated Software Systems Corporation, "Display Integrated Software System and Plotting LAnguage" software library, current version 9.2, 1505 Sorrento Valley Rd. San Diego, California 92121.

15.2. APPLICATION

15.2.1. Description.

The "PSRV" program reads an input multifile of rotated channel data files or filtered channel data files, if one of the horizontal component's data is missing and no rotation is performed. The Pseudo Relative Response Velocity (psrv) is calculated for each file in the input multifile, and the resulting data files are written to a plot file.

15.2.1.1. When to Use this Program.

The "PSRV" program should be used after the rotation program "ROTATE" has been run, to generate pseudo relative response velocity plots of the rotated radial and tangential data.

15.2.1.2. Program Functions.

A plot file is generated containing plots of the pseudo relative response velocity calculations for each of the files in the input file.

15.2.2. Operation.

Only those members of the Ground Motion and Seismic Division (7111) who are working on the project and have need to execute the "PSRV" program are authorized to access the procedure that directs the program execution. At the present time, there are three such users.

15.2.3. Equipment.

The "PSRV" program resides on the NOS/VE side of the dual state CDC CYBER 180-855. The CYBER 180-855, under NOS/VE, is configured with 6.9 million words (55.2 million bytes) of memory, and three 895 disks, each with 4 spindles, for a total of 3.6 giga-bytes of online disk space.

15.2.4. Structure.

Software for the "PSRV" program consists of FORTRAN 5 source code with calls to plot routines in a software library called "DISSPLA", both of which are resident on and executed under the control of CDC's NOS/VE.

The "PSRV" program is written in a modular fashion. Each routine performs one basic function or set of related functions, thereby providing a program that is easy to read and modify. The main program interprets the user's selection and directs the appropriate routine to verify and process the input data. When the user's input has been

dealt with appropriately, the main program determines the functions to be performed and initiates the proper routine for processing.

15.2.5. Performance.

Currently, the psrv program exists and is run interactively under control of the NOS/VE on the dual state CYBER 180-855 computer. The program can be run in batch mode if desired. Program execution is invoked from a procedure. The procedure can easily be modified to start the program in batch mode rather than in interactive mode. No changes need be made to the program to run in batch mode.

15.2.6. Inputs, Processing, and Outputs.

15.2.6.1. Inputs.

Nine parameters can be specified for the "RUN PSRV" procedure: the name of the input data file, the start time of the data, the end time of the data, the damping factor, the number of curves to plot on each plot, the number of files to process, the minimum frequency, the maximum frequency, and the name of the device to which the plot file is to be disposed. The name of the input data file is the only required parameter, however.

15.2.6.1.1. Description.

15.2.6.1.1.1. Input File Name.

The input file must be available as a disk file in the NOS/VE system. The input file name must be specified on the procedure call statement.

15.2.6.1.1.2. Start Time of the Data.

The start time must be specified in seconds; half-seconds can be specified, as in "5.5." This parameter is not required, and when a value for it is not specified on the procedure call statement, it is set to its default value of 0.0 seconds.

15.2.6.1.1.3. End Time of the Data.

The end time must be specified in seconds; half-seconds can be specified, as in "85.5." This parameter is not required, and when a value for it is not specified on the procedure call statement, it is set to its default value of 75. seconds.

15.2.6.1.1.4. The Damping Factor.

The damping factor is not required, and when a value for it is not specified on the procedure call statement, it is set to its default value of 0.05.

15.2.6.1.1.5. Number of Curves to Plot on Each Plot.

The number of curves to plot on a single plot frame must be specified as an integer number. This parameter is not required, and when a value for it is not specified on the procedure call statement, it is set to its default value of 1.

15.2.6.1.1.6. Number of Files to Process in the Input File.

This parameter is not required, and when a value for it is not specified on the procedure call statement, it is set to its default value of 10000, which will process all files in the input multifile.

15.2.6.1.1.7. The Minimum Frequency.

The minimum frequency must be specified in Herz. It must be noted that care should be taken when changing the minimum and maximum frequencies. They both should be integer powers of 10, and the maximum frequency should be 1000 times as large as the minimum frequency. This parameter is not required, and when a value for it is not specified on the procedure call statement, it is set to its default value of 0.1 Herz.

15.2.6.1.1.8. The Maximum Frequency.

The maximum frequency must be specified in Herz. It must be noted that care should be taken when changing the minimum and maximum frequencies. They both should be integer powers of 10, and the maximum frequency should be 1000 times as large as the minimum frequency. This parameter is not required, and when a value for it is not specified on the procedure call statement, it is set to its default value of 100. Herz.

15.2.6.2. Processing.

15.2.6.2.1. Initialize.

15.2.6.2.1.1. Read the Parameter List.

The program reads the input parameter values from the parameter file created by the "RUN_PSRV" procedure from the parameter values specified on the procedure call statement. When a value

for an optional parameter is not specified in the procedure call statement, the program uses the default value for the parameter.

15.2.6.2.1.2. Verify the Parameter Values.

If a parameter value is determined to be unreasonable, an error message is issued, and the parameter is set to its default value.

15.2.6.2.1.3. Determine Periods for Calculation.

The minimum and maximum periods are calculated based on the minimum and maximum frequencies.

15.2.6.2.2. The Input File.

15.2.6.2.2.1. Read the Data File.

The input data file is read from the beginning-of-file position. An ID block is read from the data file and the number of data values, the number of modification records, the minimum and maximum data values, the compression constant for converting the data, and the delta time are read from the ID block.

15.2.6.2.2.2. Prepare for the Pseudo Relative Response Velocity Calculation.

The modification records are read, skipped, and the data records are read from the input file. If the data file does not contain acceleration data, no psrv calculation is performed, and the processing continues with the next file in the input multifile. If the data records do contain acceleration data, the indices for the the data segment to process are determined from the start, stop, and delta times. The data records are then expanded from integer format to real format.

15.2.6.2.2.3. Calculate the Pseudo Relative Response Velocity.

The psrv is calculated for the frequencies between the minimum and maximum frequencies specified in the parameter list. The Newmark beta method of numerical integration is used to solve the differential equation for displacement using different values of omega. Omega represents 2π /period and ranges between 2π times the frequency minimum and maximum.

15.2.6.2.2.4. The Pseudo Relative Response Velocity Plot.

The data is plotted on a log-log-log grid. The plot is generated from the psrv calculation on the acceleration data records and is written to the plot file. A similar plot is generated for each of the data files in the input multifile.

15.2.6.3. Outputs.

The "PSRV" program creates a plot file containing the pseudo relative response velocity calculations for each of the acceleration data files within the input multifile.

15.3. PROCEDURES AND REQUIREMENTS

15.3.1. Initiation.

15.3.1.1. The Procedure.

The "PSRV" program is initiated and executed under the control of a procedure named "RUN_PSRV." "RUN_PSRV" performs all of the commands necessary to execute the "PSRV" program and dispose the plot file to the specified output device, which simplifies the user's task of running the program. The parameters that comprise the parameter input file for the program are specified on the procedure call statement. With these given values, the procedure generates the parameter input file that is used by the program, and also issues the command to dispose the plot file to the specified device.

15.3.1.2. Getting a Copy of the Procedure.

Both "RUN_PSRV" and the program source "PSRV" are resident in disk space allocated to a particular member of the Ground Motion and Seismic Division (7111). This user's disk space is called a "catalog" in the NOS/VE operating system. For a user other than the owner to access the procedure, the user name of the owner must be specified on the command that retrieves the file. The following describes the steps necessary for a user to get a copy of the procedure:

```
COPY_FILE .MLSANDE.COMMAND_CATALOG.RUN_PSRV RUN_PSRV
```

The "COPY_FILE" command will work properly only if the procedure file has been made available for access by the user requesting it. Refer to the "SCL for NOS/VE Quick Reference" manual for further information regarding copying files from another user's catalog. This command will make a copy of the procedure in the user's catalog and will name it "RUN_PSRV." When the procedure is executed, it accesses all of the files necessary to execute the program.

15.3.1.3. Invoking the Procedure.

"RUN_PSRV" creates a parameter input file for the program from the parameter list specified on the statement that invokes the procedure. The "RUN_PSRV" procedure will accept the five parameters described in section 15.2.6.1.1.1. The parameters can be specified positionally or by parameter name, and must be separated by a space or a comma. Parameters specified by a name have the following form:

```
name=value
```

where name is the parameter name and value is the parameter value.
This is the command that invokes the procedure:

```
RUN_PSRV DF=data_file NC=number_of_curves..  
          NF=number_of_files_to_process ST=start_time..  
          ET=end_time DPF=damping_factor..  
          FMN=minimum_frequency FMX=maximum_frequency..  
          DN=device_name
```

15.3.1.4. Compilation and Execution.

The procedure compiles the program source. When the source does not compile normally, a listing of the errors is displayed on the screen and the procedure is terminated. When the source compiles normally, program execution is initiated.

15.3.2. Input.

15.3.2.1. Procedure Parameters.

The "PSRV" procedure will accept nine parameters: the name of the input data file, the start time of the data, the end time of the data, the damping factor, the number of curves to plot on each plot, the number of files to process, the minimum frequency, the maximum frequency, and the name of the device to which the plot file is to be disposed.

The procedure creates a parameter input file for the program from the parameter values specified on the procedure call statement. The name of the input data file is the only required parameter. The default value for a parameter is used if an optional parameter is not entered. Refer to section 15.2.6.1. for a description of the parameters and their default values. The parameter names need not be included when the parameter values are given positionally as shown above in section 15.3.1.3.

15.3.2.2. Requirements.

15.3.2.2.1. Frequency.

The "PSRV" program should be used after the rotation program "ROTATE" has been run, to generate pseudo relative response velocity plots of the rotated radial and tangential data.

15.3.2.2.2. Origin.

The input multifile is generated by the rotation program "ROTATE."
The data files in the multifile are in the order that the "PSRV"

program expects. The program parameters are selected by the user when the procedure is initiated.

15.3.2.2.3. Medium.

The input data file is available to the program as it exists in a file on disk. The program parameters are specified on the command line entered on the terminal.

15.3.2.2.4. Restrictions.

There are no security restrictions placed on the operation of the "PSRV" program, as it resides in the open-partition CYBER 180-855 and all data is considered unclassified. Execution of the program and access to the input files is limited to only those users having knowledge about the program, the potential input files, and the user's catalog in which the files reside.

15.3.2.3. Input Formats.

15.3.2.3.1. Length.

The name of the input file may be a maximum of 14 characters in length and is not enclosed in any quote marks.

15.3.2.3.2. Format.

The input file is a multifile file, and all data files within the multifile are of the same format. Each of the data files can logically be divided into three sections: the identifier block, the modification records, and the data records. Refer to section 15.3.3.1 for a detailed description of a data file.

15.3.2.3.3. Labels.

The parameter values can be specified positionally or by parameter name, and must be separated by a space or a comma. Refer to section 15.3.1.3 for a description of the labels for the input parameters.

15.3.2.3.4. Sequence.

When the parameter values are specified without parameter names, the values must be entered in the appropriate sequence described in section 15.3.1.3. When parameter names are used with the parameter values, the order of specification on the procedure call statement is meaningless.

15.3.3. Sample Inputs.

15.3.3.1. The Input File.

The input multifile must be of a specific format. This format is the natural result of running both the filtering program "FILTER" and the rotation program "ROTATE." The format is briefly described here.

15.3.3.1.1. The Identifier Block.

The first 100-words (10 80-character lines) of a data file is the identifier (ID) block. It gives vital information regarding the data records in the file and to what event they pertain.

15.3.3.1.2. The Modification Records.

The modification (MOD) records are written to a data file by the data processing programs as the file proceeds through the data processing system. The MOD records contain information specific to the various processes performed on the data. A value in one of the words of the ID block indicates how many MOD records have been written to the data file. A MOD record, however, consists of three 80-character records. The value from the ID block must be multiplied by three to determine the correct number of MOD records.

15.3.3.1.3. The Data Records.

The data records contain the data in integer format. There are 16 data values per line, or record, of the data file. A value in one of the words of the ID block indicates how many data values are in the data file. This value is divided by 16 to determine the number of complete records of data as well as the number of data values in the last record of the file.

15.3.3.2. The Procedure Call Statement.

The following are examples of initiating the "PSRV" program.

```
RUN_PSRV data_file 1 10
```

```
RUN_psrsv data_file nf=123
```

```
Run_Psrsv DF=data_file ST=-20. NC=2 NF=5
```

15.3.4. Output.

15.3.4.1. Use.

The hard-copies of the plots are retained and stored according to event name by Division 7111.

15.3.4.2. Variations.

The format of the psrv output will vary depending on the number of curves to plot on each plot specified on the procedure call. When more than one plot is requested, the psrv plots generated are small enough to accomodate all plots requested on a single plot frame.

15.3.4.3. Destination.

15.3.4.3.1. Line Printer.

There is only one line printer available for receiving output directly from the CYBER 180-855. It is located in the Central Computing Facility in Sandia's primary technical area, Area I, in building 800. The procedure that executes this program directs all output to this line printer.

15.3.4.3.2. Plotter.

The output generated from the "PSRV" program is generally disposed to a Versatec plotter. There are several plotters available for use. One is located in the Central Computing Facility in Sandia's primary technical area, Area I, in building 800. Another, and the one to which all plots produced by this program are disposed, is located in Area I, building 806, room 166.

15.3.4.3.3. Terminal.

This program can also dispose the plot output to the terminal running the program, providing the terminal is capable of producing graphics and is supported as a graphics device. This is achieved by specifying a parameter in the procedure that executes the program.

15.3.4.4. Medium.

The plots are stored in hard-copy format, as generated by a Versatec plotter.

15.3.4.5. Disposition.

The plots generated by the "PSRV" program are stored by Division 7111 personnel in Area 1, building 806.

15.3.4.6. Sample Outputs.

15.3.4.6.1. Plot.

Figures 15.1, 15.2, and 15.3 represent sample outputs for a station generated by the "PSRV" program.

Figure 15.1

W-30 TOP AV LABQUARK
AMP X 830.00 TRACK 10 70.0 K-HZ
FREQUENCY - HZ DAMPING- 0.05
0.10

TIME 0.00 TO 75.00 SECONDS
08:53:43 1987-03-13

PSEUDO RELATIVE RESPONSE VELOCITY - CM/SEC

Figure 15.2

W-30 TOP AR LABOUARK 284 -194
AMP X 830.00 TRACK 10 52.5 K-HZ

FREQUENCY - HZ

DAMPING-

0.05

0.10

TIME 0.00 TO 75.00 SECONDS
08:54:11 1987-03-13

PSEUDO RELATIVE RESPONSE VELOCITY - CM/SEC

Figure 15.3

W-30 TOP AT LABQUARK 284 -284
AMP X 830.00 TRACK 10 40.0 K-HZ

FREQUENCY - HZ

DAMPING=

0.05

0.10

TIME 0.00 TO 75.00 SECONDS
08:54:39 1987-03-13

PSEUDO RELATIVE RESPONSE VELOCITY - CM/SEC

15.3.4.7. At Program Termination.

The program terminates normally when an end-of-information is encountered rather than an identifier block for a data file. When the "PSRV" program terminates normally, and the plots are to be disposed to the plotter, a message indicating the number of plots generated is displayed on the terminal. The command that transmits the plot file to the NOS operating system for disposition to the plotter is also displayed with all of its associated subcommands. When the terminal is specified as the plot device in the "RUN_PSRV" procedure, each plot is generated on the terminal.

15.3.5. Error and Recovery.

There are no special restart procedures. The procedure ensures its own successful termination. This allows multiple restarts of the procedure, regardless of program termination status.

If, while reading the input data file, an end-of-information is encountered anywhere other than at the beginning of the assumed next data file, the program terminates abnormally. The program and the procedure both issue error messages that are displayed on the screen, and the procedure terminates.

"VECTOR"
USER MANUAL

CHAPTER 16

"VECTOR"
USER MANUAL

16.1. GENERAL INFORMATION

16.1.1. Summary.

The "VECTOR" program calculates and plots the square roots of the sum of the squares for the vertical, radial, and tangential data records for each station for which there is data in the input file.

16.1.2. Environment.

This project is sponsored by the NNWSI project. Its development is the responsibility of the Ground Motion and Seismic Division (7111) of the Field Sciences Department (7110). Members of the project within Division 7111 are the sole users of the "VECTOR" program.

The "VECTOR" program is implemented on Sandia National Laboratories' CDC CYBER 180-855 model computer, located in the Central Computing Facility in Sandia's primary technical area, Area I, in building 800. The CYBER 180-855 operates in a dual state, which allows two operating systems to run concurrently. The "VECTOR" program runs on the NOS/VE side of the CYBER 180-855. The current operating system version is 1.1.2 level 649.

16.1.3. References.

16.1.3.1. National Bureau of Standards, "Guidelines for Documentation of Computer Programs and Automated Data Systems," Federal Information Processing Standards (FIPS) Publication #38, February, 1976.

16.1.3.2. Integrated Software Systems Corporation, "Display Integrated Software System and Plotting LAnguage" software library, current version 9.2, 1505 Sorrento Valley Rd. San Diego, California 92121.

16.2. APPLICATION

16.2.1. Description.

The "VECTOR" program reads an input multifile of rotated channel data files and calculates the three-dimensional (3-D) vector sums from the acceleration, velocity, and displacement channel files for each station. If the vertical channel data is missing for a station, two-dimensional (2-D) vector sums are calculated from the acceleration, velocity and displacement data for the radial and tangential gauges for the station. In the case that either the radial or tangential data is missing for a station, no vector sum is calculated. The vector sum data is written to both the output file and to the plot file.

16.2.1.1. When to Use this Program.

The "VECTOR" program should be used after the rotation program "ROTATE" has been run, to generate 3-dimensional and 2-dimensional vector sum plots of the data after rotation, and to create an output file containing the channels' vector sum data.

16.2.1.2. Program Functions.

16.2.1.2.1. Plot File.

A plot file is generated containing plots of the 3-D and/or 2-D vector sum calculations for each of the stations in the input multifile.

16.2.1.2.2. Output File.

An output multifile is generated containing the 3-D and/or 2-D vector sums for each of the stations in the input file.

16.2.2. Operation.

Only those members of the Ground Motion and Seismic Division (7111) who are working on the project and have need to execute the "VECTOR" program are authorized to access the procedure that directs the program execution. At the present time, there are three such users.

16.2.3. Equipment.

The "VECTOR" program resides on the NOS/VE side of the dual state CDC CYBER 180-855. The CYBER 180-855, under NOS/VE, is configured with 6.9 million words (55.2 million bytes) of memory, and three 895 disks, each with 4 spindles, for a total of 3.6 giga-bytes of online disk space.

16.2.4. Structure.

Software for the "VECTOR" program consists of FORTRAN 5 source code with calls to plot routines in a software library called "DISSPLA", both of which are resident on and executed under the control of CDC's NOS/VE.

The "VECTOR" program is written in a modular fashion. Each routine performs one basic function or set of related functions, thereby providing a program that is easy to read and modify. The main program interprets the user's selection and directs the appropriate routine to verify and process the input data. When the user's input has been dealt with appropriately, the main program determines the functions to be performed and initiates the proper routine for processing.

16.2.5. Performance.

Currently, the vector program exists and is run interactively under control of the NOS/VE on the dual state CYBER 180-855 computer. The program can be run in batch mode if desired. Program execution is invoked from a procedure. The procedure can easily be modified to start the program in batch mode rather than in interactive mode. No changes need be made to the program to run in batch mode.

16.2.6. Inputs, Processing, and Outputs.

16.2.6.1. Inputs.

Seven parameters can be specified for the "VECTOR" procedure: the name of the input data file, the name of the output data file, the volume serial number of the tape on which the output data file will eventually be written, the start time of the data, the end time of the data, a flag that specifies pseudo relative response velocity input data, and the name of the plot device to receive the plots. The name of the input data file is the only required parameter, however.

16.2.6.1.1. Description.

16.2.6.1.1.1. Input File Name.

The input file must be available as a disk file in the NOS/VE system. The input file name must be specified on the procedure call statement.

16.2.6.1.1.2. Output File Name.

The output file is generated during program execution. The output file name may be specified on the procedure call

statement. If this parameter is not specified on the procedure call statement, the program uses the default file name ROT_DATA.

16.2.6.1.1.3. Volume Serial Number of the Output Tape.

The output file will eventually be automatically written to tape. The volume serial number of the tape to write does not have to be specified until the tape-writing process has been included as a function of the "RUN_VECTOR" procedure. If this parameter is not specified on the procedure call statement, the program uses a vsn value of "000001."

16.2.6.1.1.4. The Start Time of the Data.

The start time must be specified in seconds; half-seconds can be specified, as in "5.5." This parameter is not required, and when a value for it is not specified on the procedure call statement, it is set to its default value of -25. seconds.

16.2.6.1.1.5. The End Time of the Data.

The end time must be specified in seconds; half-seconds can be specified, as in "85.5." This parameter is not required, and when a value for it is not specified on the procedure call statement, it is set to its default value of 100. seconds.

16.2.6.1.1.6. PSRV Input Data.

The parameter value must be "PSRV" (not enclosed in quote marks) to specify that the input file contains pseudo relative response velocity (psrv) data. This parameter is not required, and when a value for it is not specified on the procedure call statement, it is set to its default value of blanks, which indicates that the input file does not contain psrv data.

16.2.6.1.1.7. Plot Device.

When this parameter is specified on the procedure call statement, it must be a three-character code recognized by the NOS/VE system as a graphics output device. When this parameter is not specified, it is set to "HC1," which disposes the plot output to the Versatec plotter.

16.2.6.2. Processing.

16.2.6.2.1. Read the Parameter List.

The program reads the input parameter values from the parameter file created by the "RUN_VECTOR" procedure from the parameter values specified on the procedure call statement. When a value for an optional parameter is not specified in the procedure call statement, the program uses the default value for the parameter.

16.2.6.2.2. Separate the Channel Files for a Station.

The input file is read from the beginning-of-information position. The channel files for a station are read from the input file and written to scratch files determined by the data in the file. The acceleration, velocity, and displacement data files for the vertical gauge are written to one scratch file, the acceleration, velocity, and displacement data files for the radial gauge are written to a second scratch file, and the acceleration, velocity, and displacement data files for the tangential gauge are written to a third scratch file. When the vertical data files for a station are not in the input file, the radial and tangential data files are written to only two respective scratch files. In the case that either the radial or the tangential data files are missing, no vector sums can be calculated, and processing continues with the next station's data files.

16.2.6.2.3. The Method for Doing the Vector Sum Calculation.

The vector sum for a station can only be calculated when the vertical, radial, and tangential gauge data is present or, when the vertical gauge data is missing, the radial and tangential gauge data is present.

16.2.6.2.3.1. Determine the Data Segments for Vectorization.

The three-dimensional (3-D) vector sum is calculated using the data from the acceleration, velocity, and tangential records for a station. When the vertical data records for a station are missing, the two-dimensional (2-D) vector sum is calculated using the data from the radial and tangential records for a station. Regardless of whether 3-D or 2-D vector sums are calculated, three vector sum plots are generated for each station: one from the acceleration records, one from the velocity records, and one from the displacement records.

To calculate the three-dimensional vector sum, common start and stop times are determined for the acceleration data records. The indices for the first and last data points representing the start and stop times are also determined. The data segment for the

same period of time is then extracted from each of the acceleration data files. This defines the data to be used in the calculation of the first vector sum for the station. The same process is performed separately on the velocity and displacement data files to generate the segments of data to be used in the calculation of the second and third vector sums for the station.

The square root of the sum of the squares is performed on each set of data segments to produce the three vector sums.

16.2.6.2.4. Prepare for Vector Calculations.

16.2.6.2.4.1. Read the Data from the Scratch Files.

The identifier (id) block for the acceleration data is read from a scratch file (ref.: section 16.2.6.2.2.) and the following values are extracted: the number of data values, the number of modification records, the minimum and maximum data values, the multiplier to reconstruct the data, the start time of the data, the stop time of the data, and the data delta time. The modification records are read, skipped, and the data records are read from the scratch file. The indices and the start and stop times of the data segment are determined based on the start, stop, and delta times in the ID block and the specified start and stop times (whether user-specified or default). The data records are then expanded from integer format to real format.

For each of the scratch files, the acceleration data file is read, the start and stop times as well as the indices are determined for the data segment to be used in the vector calculation, and the data is expanded.

16.2.6.2.4.2. Extract the Data Segments.

The common start and stop times for the acceleration data records are determined such that a data segment from each channel's data records is defined for the same period of time. The data segment is extracted from each data file's acceleration data records.

16.2.6.2.5. Perform the Vector Sum Calculation.

The vector sum is calculated taking the square root of the sum of the squares for each data point in the acceleration data segments. When three-dimensional vector sums are calculated, the acceleration records for the vertical, radial, and tangential gauges are used in the calculation. When two-dimensional vector sums are calculated, the acceleration records for the radial and tangential gauges are used in the calculation.

16.2.6.2.6. Write to the Output File.

The third word in the ID block from the first acceleration data file read is replaced with a word that indicates if a 3-D or a 2-D vector sum was calculated from the station's component data. The ID block is written to the output file with the modification records from the first acceleration data file that was read. The resulting vector sum array is then written to the output file.

16.2.6.2.7. Write to the Plot File.

The plot file generated from the vector sum calculation performed on the acceleration data records is written to the plot file.

16.2.6.2.8. Calculate the Remaining Two Vector Sums.

The vector sums are also calculated for the velocity and the displacement records for the station using the data files written to the scratch files described in section 16.2.6.2.2. The processes describing reading the data records and calculating the vector sums is repeated separately for the velocity and the displacement data segments. The resulting velocity and displacement vector sums for the station are written to both the output file and to the plot file.

16.2.6.2.9. Calculate the Three Vector Sums for Each Station in the Input Multifile.

The process continues with the reading of the next station's data files for the vertical, radial, and tangential gauges from the input multifile. Processing continues as described above, beginning with section 16.2.6.2.2.

16.2.6.3. Outputs.

The "VECTOR" program creates a plot file containing the three-dimensional and two-dimensional vector sum plots calculated from the components' data for each station in the input multifile. An output file is also generated that contains the vector sum data for each component for each station.

16.3. PROCEDURES AND REQUIREMENTS

16.3.1. Initiation.

16.3.1.1. The Procedure.

The "VECTOR" program is initiated and executed under the control of a procedure named "RUN_VECTOR." "RUN_VECTOR" performs all of the commands necessary to execute the "VECTOR" program and dispose the plot file to the specified output device, which simplifies the user's task of running the program.

16.3.1.2. Getting a Copy of the Procedure.

Both "RUN_VECTOR" and the program source "VECTOR" are resident in disk space allocated to a particular member of the Ground Motion and Seismic Division (7111). This user's disk space is called a "catalog" in the NOS/VE operating system. For a user other than the owner to access the procedure, the user name of the owner must be specified on the command that retrieves the file. The following describes the steps necessary for a user to get a copy of the procedure:

```
COPY_FILE .MLSANDE.COMMAND_CATALOG.RUN_VECTOR RUN_VECTOR
```

The "COPY_FILE" command will work properly only if the procedure file has been made available for access by the user requesting it. Refer to the "SCL for NOS/VE Quick Reference" manual for further information regarding copying files from another user's catalog. This command will make a copy of the procedure in the user's catalog and will name it "RUN_VECTOR." When the procedure is executed, it accesses all of the files necessary to execute the program.

16.3.1.3. Invoking the Procedure.

"RUN_VECTOR" creates a parameter input file for the program from the parameter list specified on the statement that invokes the procedure. The "RUN_VECTOR" procedure will accept the seven parameters described in section 16.2.6.1.1. The parameters can be specified positionally or by parameter name, and must be separated by a space or a comma. Parameters specified by a name have the following form:

```
name=value
```

where name is the parameter name and value is the parameter value. This is the command that invokes the procedure:

```
RUN_VECTOR DF=data_file OF=output_file VSN=volume_serial_number
           ST=start_time ET=end_time PSRV=psrv_data..
           DN=device_name
```

16.3.1.4. Compilation and Execution.

The procedure compiles the program source. When the source does not compile normally, a listing of the errors is displayed on the screen and the procedure is terminated. When the source compiles normally, program execution is initiated.

16.3.2. Input.

16.3.2.1. Procedure Parameters.

Seven parameters can be specified for the "VECTOR" procedure: the name of the input data file, the name of the output data file, the volume serial number of the tape on which the output data file will eventually be written, the start time of the data, the end time of the data, a flag that specifies pseudo relative response velocity input data, and the name of the plot device to receive the plots.

The procedure creates a parameter input file for the program from the parameter values specified on the procedure call statement. The name of the input data file is the only required parameter. The default value for a parameter is used if an optional parameter is not entered. Refer to section 16.2.6.1. for a description of the parameters and their default values. The parameter names need not be included when the parameter values are given positionally as shown above in section 16.3.1.3.

16.3.2.2. Requirements.

16.3.2.2.1. Frequency.

The "VECTOR" program should be run after the rotation program "ROTATE" has been run, to generate plots of the vector sums of the data for each station in the input data file.

16.3.2.2.2. Origin.

The input multifile is generated by the filtering program "ROTATE." The data files in the multifile are in the order that the "ROTATE" program produces. The program parameters are selected by the user when the procedure is initiated.

16.3.2.2.3. Medium.

The input data file is available to the program as it exists in a file on disk. The program parameters are specified on the command line entered on the terminal.

16.3.2.2.4. Restrictions.

There are no security restrictions placed on the operation of the "VECTOR" program, as it resides in the open-partition CYBER 180-855 and all data is considered unclassified. Execution of the program and access to the input files is limited to only those users having knowledge about the program, the potential input files, and the user's catalog in which the files reside.

16.3.2.3. Input Formats.

16.3.2.3.1. Length.

The name of the input file may be a maximum of 14 characters in length and is not enclosed in any quote marks. The name of the output file may be a maximum of 14 characters in length and is not enclosed in any quote marks.

16.3.2.3.2. Format.

The input file is a multifile file, and all data files within the multifile are of the same format. The output file is also a multifile file, and all data files within the multifile are of the same format as the input multifile. Each of the data files can logically be divided into three sections: the identifier block, the modification records, and the data records. Refer to section 16.3.3.1 for a detailed description of a data file.

16.3.2.3.3. Labels.

The parameter values can be specified positionally or by parameter name, and must be separated by a space or a comma. Refer to section 16.3.1.3 for a description of the labels for the input parameters.

16.3.2.3.4. Sequence.

When the parameter values are specified without parameter names, the values must be entered in the appropriate sequence described in section 16.3.1.3. When parameter names are used with the parameter values, the order of specification on the procedure call statement is meaningless.

16.3.3. Sample Inputs.

16.3.3.1. The Input File.

The input multifile must be of a specific format. This format is the natural result of running the rotation program "ROTATE." The format is briefly described here.

16.3.3.1.1. The Identifier Block.

The first 100-words (10 80-character lines) of a data file is the identifier (ID) block. It gives vital information regarding the data records in the file and to what event they pertain.

16.3.3.1.2. The Modification Records.

The modification (MOD) records are written to a data file by the data processing programs as the file proceeds through the data processing system. The MOD records contain information specific to the various processes performed on the data. A value in one of the words of the ID block indicates how many MOD records have been written to the data file. A MOD record, however, consists of three 80-character records. The value from the ID block must be multiplied by three to determine the correct number of MOD records.

16.3.3.1.3. The Data Records.

The data records contain the data in integer format. There are 16 data values per line, or record, of the data file. A value in one of the words of the ID block indicates how many data values are in the data file. This value is divided by 16 to determine the number of complete records of data as well as the number of data values in the last record of the file.

16.3.3.2. The Procedure Call Statement.

The following are examples of initiating the "VECTOR" program.

```
RUN_VECTOR data_file out_file
```

```
RUN_vector data_file vsn=12314
```

```
Run_Vector DF=data_file ST=-10 DN=tk4
```

"VECTOR"
USER MANUAL

16.3.4. Output.

16.3.4.1. Use.

The hard-copies of the plots are retained and stored according to event name by Division 7111. The output file containing the data necessary to re-generate the vectored data plots is also stored in a disk file belonging to a member of Division 7111.

16.3.4.2. Destination.

16.3.4.3.1. Line Printer.

There is only one line printer available for receiving output from the CYBER 180-855. It is located in the Central Computing Facility in Sandia's primary technical area, Area I, in building 800. The procedure that executes this program directs all output to this line printer.

16.3.4.2.2. Plotter.

The output generated from the "VECTOR" program is generally disposed to a Versatec plotter. There are several plotters available for use. One is located in the Central Computing Facility in Sandia's primary technical area, Area I, in building 800. Another, and the one to which all plots produced by this program are disposed, is located in Area I, building 806, room 166.

16.3.4.2.3. Terminal.

This program can also dispose the plot output to the terminal running the program, providing the terminal is capable of producing graphics and is supported as a graphics device. This is achieved by specifying a parameter in the procedure that executes the program.

16.3.4.3. Medium.

The plots are stored in hard-copy format, as generated by a Versatec plotter. The output file containing the data necessary to re-generate the filtered data plots is stored in the file space of a member of Division 7111.

16.3.4.4. Disposition.

The plots generated by the "VECTOR" program are stored by Division 7111 personnel in Area 1, building 806.

"VECTOR"
USER MANUAL

16.3.4.5 Sample Outputs.

16.3.4.5.1. Plot.

Figures 16.1, 16.2, and 16.3 represent sample outputs for a station generated by the "VECTOR" program.

16.3.4.5.2. Output Data File.

Figure 16.4 represents the format of the output file from the "VECTOR" program.

Figure 16.1

AMP X 830.00	DEL: .005
TRACK 10	MIN: .9621E-05
70.0 KHZ	MAX: .3703E-01
1987-03-13	AZM: 194.6659
11:13:13	RNG: 55362.46
	VSN: 000001

Figure 16.2

AMP X 830.00
TRACK 10
70.0 KHZ
1987-03-13
11:13:36

DEL: .005
MIN: .4946E-06
MAX: .9419E-02
AZM: 194.6659
RNG: 55362.46
VSN: 000001

Figure 16.3

AMP X 830.00
TRACK 10
70.0 KHZ
1987-03-13
11:13:57

DEL: .005
MIN: .8300E-04
MAX: .3239E+00
AZM: 194.6659
RNG: 55362.46
VSN: 000001

Figure 16.4

SCAN OF TAPE scan_data

FILE NO.	DATA ID	STATION	ORIENT	EVENT	ROTATE	APP	TR FREQ	START	END
1	FILTER 2	W-30 TOP	AV	LABOUARK		830.00	TR1070.0	-20.0000	99.9980
2	INTEG 2	W-30 TOP	AV	LABOUARK		830.00	TR1070.0	-20.0000	99.9980
3	INTEG2 2	W-30 TOP	AV	LABOUARK		830.00	TR1070.0	-20.0000	99.9980
4	ROTATE 3	W-30 TOP	AR	LABOUARK	284 -194	830.00	TR1052.5	-20.0000	99.9980
5	INTEG 3	W-30 TOP	AR	LABOUARK	284 -194	830.00	TR1052.5	-20.0000	99.9980
6	INTEG2 3	W-30 TOP	AR	LABOUARK	284 -194	830.00	TR1052.5	-20.0000	99.9980
7	ROTATE 3	W-30 TOP	AT	LABOUARK	284 -284	830.00	TR1040.0	-20.0000	99.9980
8	INTEG 3	W-30 TOP	AT	LABOUARK	284 -284	830.00	TR1040.0	-20.0000	99.9980
9	INTEG2 3	W-30 TOP	AT	LABOUARK	284 -284	830.00	TR1040.0	-20.0000	99.9980

DATA WAS PROCESSED IN ACCORDANCE WITH ETP-7111-1.

SCAN - NORMAL TERMINATION.
9 FILES PROCESSED.

16.3.4.6. At Program Termination.

The program terminates normally when an end-of-information is encountered rather than an identifier block for a data file. When the "VECTOR" program terminates normally, and the plots are to be disposed to the plotter, a message indicating the number of plots generated is displayed on the terminal. The command that transmits the plot file to the NOS operating system for disposition to the plotter is also displayed with all of its associated subcommands. When the terminal is specified as the plot device in the "RUN_VECTOR" procedure, each plot is generated on the terminal.

16.3.5. Error and Recovery.

There are no special restart procedures. The procedure ensures its own successful termination. This allows multiple restarts of the procedure, regardless of program termination status.

If, while reading the input data file, an end-of-information is encountered anywhere other than at the beginning of the assumed next data file, the program terminates abnormally. The program and the procedure both issue error messages that are displayed on the screen, and the procedure terminates.

DISTRIBUTION LIST

Office of Civilian Radioactive (14)

Waste Management

U.S. Department of Energy

Forrestal Bldg.

Attn: C.E. Kay, Acting Director (RW-1)

D. H. Alexander (RW-232)

C. Bresee (RW-22)

V. J. Cassella (RW-222)

J. J. Fiore (RW-221)

M. W. Frei (RW-231)

B. G. Gale (RW-223)

R. W. Gale (RW-40)

T. H. Isaacs (RW-20)

Allen Jelacic (RW-233)

S. H. Kale (RW-20)

J. P. Knight (RW-24)

Gerald Parker (RW-241)

Ralph Stein (RW-23)

Washington, D.C. 20585

U.S. Nuclear Regulatory Commission (3)

Division of Waste Management

Attn: Chief, Repository Projects Branch

NTS Section Leader,

Repository Projects Branch

Document Control Center

Washington, D.C. 20555

U.S. Department of Energy (20)

Nevada Operations Office

Attn: P. K. Fitzsimmons, Director,

Health Physics & Environmental
Division

J. L. Fogg, Technical Information
Office (12)

Carl P. Gertz, Project Manager,
Waste Management Project Office (6)

C. L. West, Director,
Office of External Affairs

P.O. Box 98518

Las Vegas, NV 89193-8518

A. T. Tamura

Science and Technology Division

Office of Scientific and Technical
Information

U.S. Department of Energy

P.O. Box 62

Oak Ridge, TN 37831

J. O. Neff, Manager
Salt Repository Project Office
U.S. Department of Energy
110 North 25 Mile Avenue
Hereford, TX 79045

S. A. Mann, Manager
Crystalline Rock Project Office
U.S. Department of Energy
9800 South Cass Avenue
Argonne, IL 60439

Lawrence Livermore National Laboratory (2)
Attn: K. Street, Jr., MS L-209
L. D. Ramspott,
Technical Project Officer
for NNWSI, MS L-204 (3)
P.O. Box 808
Livermore, CA 94550

Fenix & Scisson, Inc. (2)
Attn: J. A. Cross, Manager
Las Vegas Branch
R. L. Bullock, Technical
Project Officer for NNWSI
P.O. Box 93265
Mail Stop 514
Las Vegas, NV 89193-3265

V. M. Glanzman
U.S. Geological Survey
P.O. Box 25046
913 Federal Center
Denver, CO 80225

P. T. Prestholt
NRC Site Representative
1050 East Flamingo Road
Suite 319
Las Vegas, NV 89109

Science Applications International (3)
Corporation
Attn: M. E. Spaeth,
Technical Project Officer
for NNWSI
SAIC-T&MSS Library (2)
101 Convention Center Drive
Suite 407
Las Vegas, NV 89109

Science Applications International
Corporation
Attn: W. S. Twenhofel, Consultant
820 Estes Street
Lakewood, CO 89215

Science Applications International
Corporation
Attn: T. G. Barbour
1626 Cole Blvd., Suite 270
Golden, CO 80401

Holmes and Narver, Inc.
Attn: A. E. Gurrola, General Manager,
Energy Support Division
P.O. Box 93838
Mail Stop 580
Las Vegas, NV 89193-3838

Holmes and Narver, Inc.
Attn: J. P. Pedalino, Technical
Project Officer for NNWSI
101 Convention Center Drive
Suite 860
Las Vegas, NV 89109

Los Alamos National Laboratory
Attn: D. T. Oakley, Technical
Project Officer for NNWSI (4)
P.O. Box 1663
N-5, Mail Stop J521
Los Alamos, NM 87545

U.S. Geological Survey (6)
Attn: L. R. Hayes, Technical
Project Officer for NNWSI
P.O. Box 25046
421 Federal Center
Denver, CO 80225

Nuclear Waste Project Office(4)
State of Nevada
Attn: C. H. Johnson,
Technical Program Manager
R. R. Loux, Jr., Executive
Director (3)
Evergreen Center, Suite 252
1802 North Carson Street
Carson City, NV 89701

ONWI Library
Battelle Columbus Laboratory
Office of Nuclear Waste Isolation
505 King Avenue
Columbus, OH 43201

Roy F. Weston, Inc.
Attn: W. M. Hewitt,
Program Manager
Technical Information Center
935 L'Enfant Plaza, Southwest
Suite 800
Washington, D.C. 20024

H. D. Cunningham, General Manager
Reynolds Electrical & Engineering Co.
P.O. Box 14400
Mail Stop 555
Las Vegas, NV 89114

Reynolds Electrical & Engineering Co. (2)
Attn: D. L. Fraser,
General Manager
Vincent Gong, Technical Project
Officer for NNWSI
P.O. Box 98521
Mail Stop 555
Las Vegas, NV 89193-8521

T. Hay, Executive Assistant
Office of the Governor
State of Nevada
Capitol Complex
Carson City, NV 89710

John Fordham
Desert Research Institute
Water Resources Center
P.O. Box 60220
Reno, NV 89506

Dr. Martin Mifflin
Desert Research Institute
Water Resources Center
2505 Chandler Avenue
Suite 1
Las Vegas, NV 89120

Department of Comprehensive
Planning
Clark County
225 Bridger Avenue, 7th Floor
Las Vegas, NV 89155

Lincoln County Commission
Lincoln County
P.O. Box 90
Pioche, NV 89043

Community Planning & Development
City of North Las Vegas
P.O. Box 4086
North Las Vegas, NV 89030

City Manager
City of Henderson
Henderson, NV 89015

E. P. Binnall
Field Systems Group Leader
Building 50B/4235
Lawrence Berkeley Laboratory
Berkeley, CA 94720

B. J. King, Librarian (2)
Basalt Waste Isolation Project Library
Rockwell Hanford Operations
P.O. Box 800
Richland, WA 99352

J. H. Anttonen
Deputy Assistant Manager for
Commercial Nuclear Waste
Basalt Waste Isolation Project Office
U.S. Department of Energy
P.O. Box 550
Richland, WA 99352

Prof. S. W. Dickson
Department of Geological Sciences
Mackay School of Mines
University of Nevada
Reno, NV 89557

Planning Department
Nye County
P.O. Box 153
Tonopah, NV 89049

Economic Development Department
City of Las Vegas
400 East Stewart Avenue
Las Vegas, NV 89101

Director of Community Planning
City of Boulder City
P.O. Box 367
Boulder City, NV 89005

Commission of the European Communities
200 Rue de la Loi
B-1049 Brussels
BELGIUM

J. R. Rollo
Deputy Assistant Director for
Engineering Geology
U.S. Geological Survey
106 National Center
12201 Sunrise Valley Dr.
Reston, VA 22092

Eric Anderson
Mountain West Research-Southwest, Inc.
Phoenix Gateway Center
432 North 44 Street
Suite 400
Phoenix, AZ 85008-6572

Judy Foremaster (5)
City of Caliente
P.O. Box 158
Caliente, NV 89008

Elaine Ezra
NNWSI GIS Project Manager
EG&G Energy Measurements, Inc.
P.O. Box 1912
Mail Stop H-02
Las Vegas, NV 89125

Leslie Jardine
Project Manager
Bechtel National, Inc.
P.O. Box 3965
San Francisco, CA 94119

R. Harig
Parsons, Brinckerhoff, Quade &
Douglas, Inc.
1625 Van Ness Avenue
San Francisco, CA 94109-3678

Librarian
Los Alamos Technical Associates, Inc.
P.O. Box 410
Los Alamos NM 87544

Internal Distribution:

6300 R. W. Lynch
6310 T. O. Hunter
6310 NNWSICF
6310 23/12412/7110/REP/QA1 (XREF)
6310 100/12412/SAND/Q1
6311 A. L. Stevens
6311 C. Mora
6312 F. W. Bingham
6313 T. E. Blejwas
6313 B. M. Schwartz (2)
for DRMS Files
55/F08-04/07/87
55/F08-06/30/66
6314 J. R. Tillerson
6315 S. Sinnock
6316 R. B. Pope
6332 WMT Library (20)
6430 N. R. Ortiz
3141 S. A. Landenberger (5)
3151 W. L. Garner (3)
8024 P. W. Dean
3154-3 C. H. Dalin (28)
for DOE/OSTI
7311 M. Sanders
7111 J.S. Phillips
6311 C.V. Subramanian