USER REFERENCE MANUAL

KGS MICROCOMPUTER-BASED

SEISMIC PROCESSING SYSTEM

(Version 2.0 - 11/17/88)

Kansas Geological Survey
Open-file Report

Disclaimer

Revision Dates
Version 1.x : 4/28/87, 5/14/87, 7/9/87, 8/1/87, 9/3/87,
10/9/87, 10/20/87, 12/21/87, 3/18/88
Version 2.x : 10/26/88, 11/17/88

TABLE OF CONTENT Page

FIGURE CAPTIONS Page

0. SYSTEM COMPONENTS

The current KGS microcomputer-based seismic processing system consists of Zenith Data Systems 386 model 80 microcomputer. The IBM PC/AT compatible microcomputer is equipped with an 80387 co-processor and a high-resolution (640x350 pixels) Enhanced Graphics Adapter (EGA) Card.

System Requirements

For general system environment, the following hardware and software are mandatory for the correct operation of the KGS seismic processing package.

- IBM XT, AT or 386 compatible computer, running MicroSoft DOS 3.1 or MicroSoft OS/2 Ver 1.0.
- 640 Kbyte of memory
- IBM compatible Graphic printer, connected to LPT1
- CGA, EGA or VGA color video interface card w/256K RAM, and color monitor
- 1 or 2 1.2M floppy disk drives
- 10 Mbyte or more fixed disk storage (As a practical limit 20 Mbyte should be considered minimum.)

Highly recommended additional hardware:

- 8087,80287, or 80387 math co-processor.
- 40 Mbyte or more 28 millisecond fixed disk.
- 5 Meg RAM memory if using OS/2.

OS/2 notes:
All NON-Graphics programs can be run under OS/2. Graphics programs (PLOT,VELP,FMAIN) can be run under the OS/2-MS/DOS compatibility window as can non-graphic programs.

1. SEISMIC DATA ORGANIZATION

All seismic data are stored in a 16-bit fixed point format, ie. the sample values of all seismic-traces are between -32768 and +32767 (2-byte integer).

1.1 Basic Data Organization

All seismic data in mass-storage (hard disk) are organized as a series of traces.

Each trace consists of a 120-word fixed length trace-header followed by a variable length data series (but data size must be fixed within the dataset). The length of the data series is dependent on the number of data samples. The first data sample value is in word 121 of the trace.

All programs preserve and/or update the trace header of the input data.

1.2 Trace Header Description

The following trace header words are currently used for specification of parameters and control of processing sequences:

16-bit Word Number	Description
1	Data type: 0 = Raw field 1 = CDP gather 2 = CDP stacked 3 = Record order (Record number index and Trace number index based on values in trace header word 3 and 4, respectively) 4 = Velocity-scan data
2	Total recording channels
3	Trace Header Word of RECORD Number for this data set where 8 = common recording channel number 12 = common depth point 19 = common offset 86 = common receiver station number 87 = common source station number 92 = common source sequence number
4	Trace Header Word of TRACE Number within each record (0 = as input order of seismic input data to be sorted)
5	Trace direction flag for sorted traces within each record 1 = ascending -1 = descending
6	Original Field record number
8	Recording channel number
10	Repeated shot number (at the same station)
12	CDP number
14	Trace number within each record
15	Trace identification code: 1 = seismic data 2 = dead 9 = velocity flag
16	Number of vertically summed traces yielding this trace
17	Number of horizontally summed traces yielding this trace
19	Offset (distance from source to receiver) after multiplied by word 35
21**	Receiver group elevation
23**	Source elevation
27**	Datum elevation

35 Multiplication factor for horizontal distance

50* Source static correction (ms)
51* Receiver group static correction (ms)
52* Total static correction (ms) that HAS BEEN applied to this trace (zero if no static has been applied)

55 Recording delay time (ms)

58 Number samples in this trace
59 Sample interval in micro-seconds for this trace

70 Analog low-cut frequency (Hz)
71 Analog high-cut frequency (Hz)

75 Applied digital low-cut frequency (Hz)
76 Applied digital high-cut frequency (Hz)

82 Minimum receiver station number
83 Maximum receiver station number
84 Minimum source sequence number
85 Maximum source sequence number

86 Receiver station number for this trace
87 Source station number for this trace

88 Last trace flag: 0 = not last trace. 1 = last trace.

89* Surface consistent residual receiver-static (in number of SAMPLES) that HAS BEEN applied to this trace
90* Surface consistent residual source-static (in number of SAMPLES) that HAS BEEN applied to this trace

92 Source sequence number

93 Processing history file flag: 0 = No history
non zero = number of characters in file name to follow.

94-120 Reserved for processing history file name. Packed ASCII. Two ASCII characters per word.

* Convention for static corrections :
POSITIVE value implies static shift (DOWN) away from zero-time,
NEGATIVE value implies static shift (UP) toward zero-time.

**Elevation can be either absolute (ie. positively above sea level) or relative (with reference to fixed altitude). In both cases, the orientation is such that higher elevation is positive. Therefore, increasing depth is indicated by the smaller value for elevation.

NOTE: ms = milli-seconds

2. PROCESSING PROCEDURES

Diagram of processing sequence is shown in a figure 1.

Processing methods are divided into two main categories: interactive and card-control.

The first method (interactive) is used mainly for "data analysis" (eg. velocity analysis, spectral analysis, etc.). It is intended for quick and accurate analysis in order to design parameters used later in "data processing".

The second method (card-control) is used mainly with "data processing" (eg. sorting, NMO, stack, etc.) to produce a final stacked seismic section.

Current interactive processings include:

- data conversion into KGS modified SEG-Y format (BISCONV)
- dump chosen trace header words (TRHD)
- velocity calculation from reflection events (VELP)
  (can apply to either field-record data or CDP data)
- velocity-scan analysis (VSCN)
  (can apply only to CDP data)
- spectral analysis, filtering, and deconvolution (FMAIN)
- data display (PLOT)

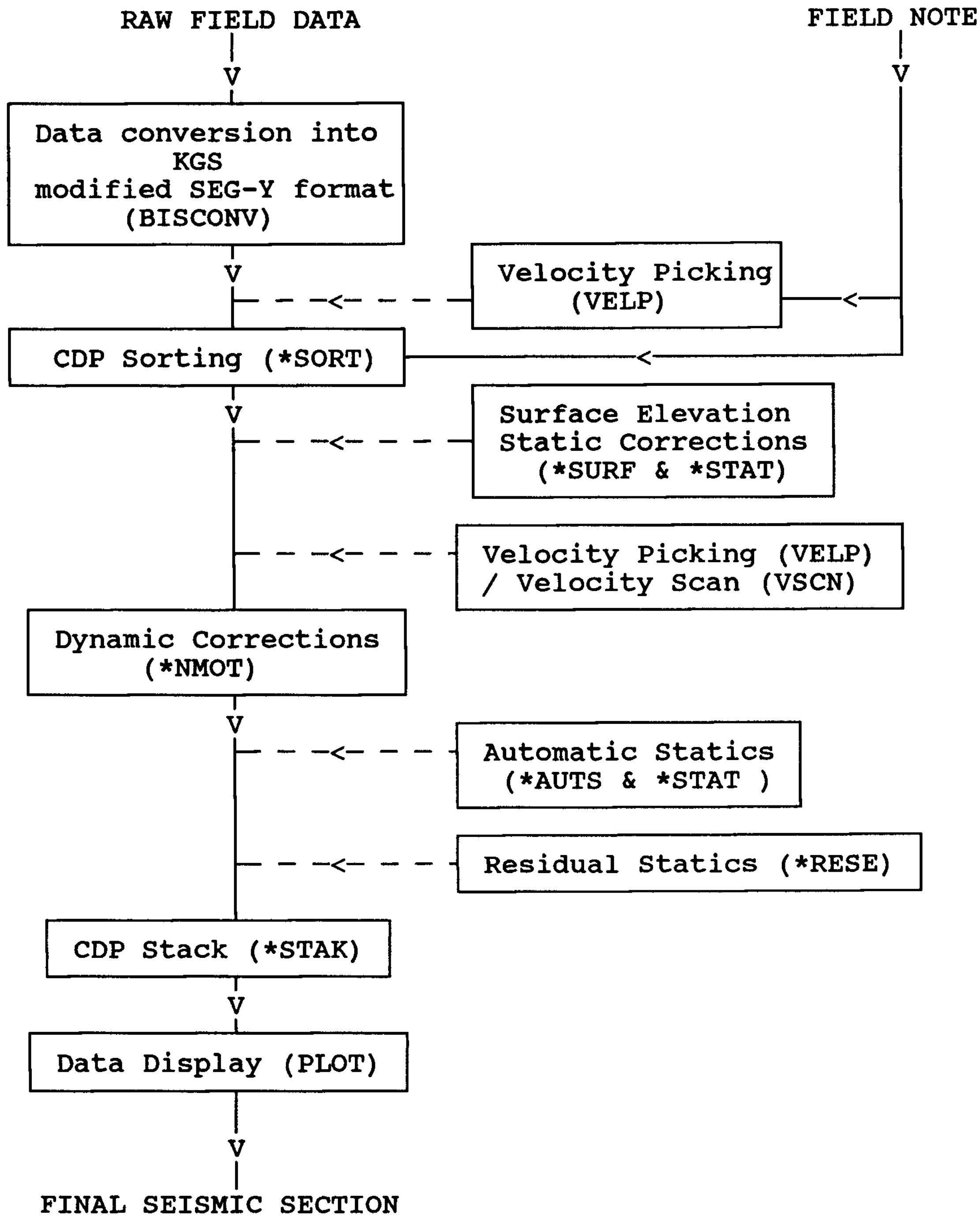
Current card-control processings include:

- Trace Sorting (*SORT, *RSRT)
- Trace Editing (*EDMT, *EDFM, *EDKL, *AUED)
- Dynamic Correction (*NMOT)
- Static Correction (*SURF, *AUTS, *STAT, *RESE)
- CDP Stacking (*STAK)
- AGC Scaling (*SCAL)

Section 3 describes various available interactive programs. They are designed to perform a specific single task or processing procedure interactively (rather than a multi-operation program).

Section 4 describes a card-control processing software package (called "SEIS"). It is designed to be a multi-operation processing system.

Section 5 describes an interactive software package (called "FMAIN"). It has capabilities to perform spectral analysis, filtering, and deconvolution both in graphically interactive mode and batch-processing mode.



NOTE: Can apply Filtering, Spectrum Analysis, and Deconvolution (FMAIN) at any stage after BISCONV.

Figure 1. Processing sequence diagram.
Procedure names are in parentheses.
Procedure names with "*" are card-control processing.
Others are interactive.

3.0 INTERACTIVE PROCESSING

3.1 PLOT (Plot/view seismic data Version 4.0)

Input data : KGS seismic data files.

PLOT USAGE

```
 PLOT ?  Gives this help list
 PLOT  (no parameters) Interactive questions
 PLOT fname start end dump bits ips norm stime tspace sdel outfile

 Where:
 FNAME  = File name to plot. (Path optional)
 START  = Starting record number to plot, 0=first in file.
 END    = Ending record number to plot, 0=plot entire file.
 DUMP   = Auto-printing, 0=wait before printing,1=auto-print.
*IPS    = Vertical Inches/Second. 0=1:1 printing.
 NORM   = Plot normalization. 0=Plot normal OFF, 1=ON.
 STIME  = Starting time (ms) of plot.
*TSPACE = Trace separation in traces/inch on hardcopy.0=20T/inch
 SDEL   = Plot normalization scan-delay time in Ms.
         (Only needed if Plot normal is on or re-direction
needed)
 OUTFILE= File name where graphics image is to be re-directed.
         (Only available with command line invocation)
```

Any number of the parameters may entered.
Values not given on the command will cause the program
to become interactive at the missing parameter.

* Denotes changes with version 4
  Bits question has been removed.

The PLOT program is the user's way of visually examining the results of data processing or the original data itself. Plot is designed to operate with CGA, EGA or VGA video systems and an Epson graphics or IBM graphics printer. Because of the limited nature of the various video display modes only 200, 350, or 480 (CGA,EGA,VGA respectively) samples of the data can be seen on the CRT. With a standard 8 inch printer, up to 960 samples per trace may be printed. Systems not equipped with graphics video hardware may still use it to produce printed graphics.

Plot will also display any processing history that maybe present with the data. See "processing history handling" in card control section for more information on processing history.

(This space intentionally left blank)

Plot has two modes of operation. Command Line Invocation (CLI) and Interactive Mode (IM). The syntax for using PLOT in CLI mode is:

```
PLOT FNAME START END DUMP BITS IPS NORM STIME TSPACE SDEL OUTFILE
```

FNAME:
FNAME is the name of a data file to be plotted. FNAME maybe any valid MS-DOS file name and may include a path if needed. The data must be in standard KGS-SEGY format. Plot will inspect the file to make sure it is in the correct format. If the file is not in the correct format plot reports FILE I/O ERROR 1 or FILE I/O ERROR 2. If the file does not exist, then PLOT prints FILE I/O ERROR 0. These checks occurs only when the file is first opened, if the file passes these tests but is actually corrupt, unpredictable results may occur.

START:
START is record number to begin the plot with. In the case of field data this will be a source sequence number (SSN). In the case of CDP sorted data this will be the CDP number to start with. Entering a 0 in the CLI mode will cause PLOT to start with the first number found. Entering a 0 in IM, however, will cause PLOT to search for a record with the number 0. When specifying a starting record number deep within the data set, expect a noticeable delay while PLOT searches for the requested record.

END:
END is the record number where PLOT should stop plotting (including that number.) In the case of field data this number will represent the last SSN to plot, in CDP data, the last CDP to plot. Entering a 0 in CLI mode instructs PLOT to plot all data in the file. Entering a 0 in IM will cause the plotting to stop if a record with the number 0 is encountered.

DUMP:
DUMP is a boolean (True/False) parameter that controls the graphic output generated by PLOT. If this value is a 1 (true) then PLOT will automatically send its graphics data to the printer. (Or re-direct as dictated by OUTFILE, see below.) If this value is 0 (false) then PLOT prompts the user after each screen full of graphics with the message: P=Print C=Continue. The user then must type a P or a C . Typing a P will cause the image to be printed, a C will cause the next screen full of data to appear. CTRL-C may also be typed to abort PLOT, and return to MS-DOS.

IPS:

Inches-per-second (IPS) controls the vertical (time) scale of the plot. By correct manipulation of this value the user can control the size of the printed output. Use of this parameter requires knowledge of the data's trace length in seconds. Assume for example, that the data to be plotted has a sample rate of 1 ms/sample and there are 500 samples. This yields a trace length (TL) of .5 seconds (TL = .001 Seconds/Sample * 500 samples = 0.5 Seconds.) Choosing an IPS value of 10 would produce a plot 5 inches tall. (10 inches/seconds * TL = 5 inches.) It is important to keep in mind the physical limits of the printer when picking an IPS value. If the value 20 had been chosen for the above example then Plot would try to expand the data to a total of 10 inches. (20 Inches/Second * TL = 10 inches.) However a standard printer has only 8 inches of printing area! In this case Plot will print only the first 400 ms of the data.
(8 inches / 20 Inches/Second = .4 Seconds = 400 ms.)

As a final example assume that operator wants ALL of the data printed in 4 inches. Choosing an IPS value of 8 would do this.
(4 Inches / TL = 8 Inches/Second.)

If IPS is set to 0 then Plot will compute a default Inches/Second value, such that, each dot printed represents one sample of data. This number is a function of the resolution of the printer, which for an Epson is 120 Dots/Inch. Using the example from above, Plot would choose a default IPS value of 8.333. The equation to find this is: (500 samples / 120 Dots/inch)/TL. If the number of samples exceeds the printing capacity of the printer, i.e. more than 960 samples on an 8 inch Epson (120 Dots/Inch * 8 Inches = 960 dots), Plot will choose a default IPS value such that all of the data will fit on the page.

NOTE: When choosing a value for IPS other than default, Plot is forced to interpolate the data. This interpolation allows Plot to either expand or shrink the apparent number of samples. When Plot presents this data to the CRT for display, only as many samples as the CRT's graphics hardware will allow can be shown. (200 for CGA , 350 for EGA and 480 for VGA.)

NORM:

NORM is a boolean parameter controlling the application of a SINGLE trace normalizing function. If the NORM is a 1 (ON) then before each trace is plotted it is scanned for the largest amplitude value in that trace. Each value in the trace is then divided by this largest value, effectively normalizing all values to the maximum. If NORM is a 0 (OFF) then before the data plotted it is first divided by the maximum value found in the first 65000 samples rather than the maximum of each trace. In either case after the data is normalized it is then scaled. This scaling is a function of the trace spacing (TSPACE) and the Amplitude Adjust Factor. (See TSPACE, Customizing Plot, and SDEL.)

STIME:
The STIME parameter controls where in TIME (in milliseconds) the plot will begin. If this value is 0 then the display starts at sample 1 for each trace and plots as many as it can. This parameter is generally only useful for those times when one only wants to view the data on the screen and the event of interest is near the end of data and not being shown on the screen.

TSPACE:
The TSPACE parameter controls the spacing between the traces as they are printed. This number is in Traces/Inch. Choosing a value of 10, for instance, will produce a plot with approximately 10 traces for every inch of paper printed. The default value is 20 traces/inch. Traces/inch values much above 24 quickly degrade in appearance as the resolution of the printer is limited. The final width of the plot in inches can be estimated in advance by the following :
Length of plot on an Epson printer with 144 Dot/Inch resolution = (Number of traces to be plotted x 144 Dots/Inch) / Traces/Inch.

NOTE: Since video monitors vary in size the Traces/Inch as seen on the users CRT will look much different than that which is printed.

SDEL:
The SDEL parameter (ScanDELay time) controls the operation of the normalize function. Normally the normalize function scans all samples in the trace to find the largest one, with which it will normalize the data. The SDEL represents how far down in time (Milliseconds) to skip before searching for this largest value. The effect of this is to increase the amplitude of smaller level signals. This parameter is only meaningful if the NORM parameter is 1, and may be skipped. (Note: Enter a 0 in this parameter even if Normalizing is off and you wish to enter an OUTFILE name.)

OUTFILE:
Outfile is a file name where the output from PLOT is to be re-directed. If this parameter is left blank then the default file LPT1 is used. LPT1 is the MS-DOS name for Line PrinTer 1. If for instance you were using a serial printer on COM port 2, you could place the name COM2 in this parameter and the graphics output would be directed to the serial printer. Valid MS-DOS device names include: LPT1, LPT2, COM1, COM2, and AUX. Any valid MS-DOS file name may also be entered. Using this, one can create a data file that can be copied from disk to printer at a later time. Valid MS-DOS path names are also allowed. The file is opened in APPEND mode, so if a file with that name is already present, data is appended to the end of it.

Customizing Plot

When started, Plot looks for a file named PLOT.CFG.
If this file is present, Plot will read the contents and change some of it's default settings. The format of PLOT.CFG is as follows:

Density	(0)
Dots/line	(960)
Vertical resolution	(120)
Horizontal resolution	(144)
Amplitude Adjust Factor	(0)
Trace type	(0)

The numbers in parenthesis are the default values chosen if PLOT.CFG is not present.

Density is the number 0 or 1. 0 means low density graphics, 1 means high density graphics. Low density graphics print faster but with lower quality than high density graphics.

Dots/line tells Plot how many printable dots per line there are for the attached printer. Standard 8 inch printers will have 960 dots. A wide carriage printer will generally have 1632 printable dots. Setting this value larger than the printer can actually handle may cause unexpected results.

Vertical resolution is the number of dot/inch in the vertical (time) scale. This number is used to compute the default Inches-Per-Second (IPS) value. This number should be set to 120 for standard graphics printers.

Horizontal resolution is the number of dots/inch in the horizontal (distance) scale. This number is used to compute the default Traces/Inch (TSPACE). This number should be set to 144 for standard graphics printers. Horizontal and Vertical resolution values relate to the way in which the data is plotted. Since Plot rotates data 90 degrees before printing vertical refers to the 'width' of the paper and horizontal the 'length' of the paper.

Amplitude Adjust Factor (AAF) allows the user to adjust the size of the wiggle traces that PLOT prints. As discussed above, (see NORM) before plotting Plot normalizes the data by dividing each sample by the maximum value found in the data. This results in numbers ranging from -1 through 1. Before this data can be plotted it must be scaled up by some factor. This scale factor is based upon the trace spacing chosen (See TSPACE.) The scale factor Plot chooses is sufficient for most data, however there are times when it is not. AFF can adjust the amplitude of the traces up or down. When AFF is 0, the standard trace deflection is used. When AFF is greater than 0, the amplitude of the traces increases. For each increment of AFF the amplitude of the traces is DOUBLED. For example a value of 2 would produce a plot with 4 times the amplitude of a plot using an AFF of 0. For values less than 0 the amplitude of the traces decreases. For each decrement of AFF the amplitude of the traces is HALVED. For example a value of -2 would produce a plot with 1/4 (.25) the amplitude of a plot using and AFF of 0. Only INTEGER values are allowed for AFF in the range of -5 to 5.

TRACE TYPE is a flag, 0 or 1, controlling the type of line that is drawn during plotting. When TRACE TYPE is a 0 PLOT uses a variable area fill when drawing traces. When TRACE TYPE is a 1 only simple wiggle traces are drawn. (Using wiggle trace, TRACE TYPE = 1, will increase the life of your printer ribbon!)

Each line must end with a carriage return - line feed (CR-LF) pair.
Optionally the user may add text after each value. The only requirement is that there are 1 or more spaces between the value and comment field. For example below is a valid PLOT.CFG file.

```
1    print in high density
1632 Epson FX-286e wide carriage printer
120  vertical resolution for an Epson printer
144  horz resolution for an Epson printer
-1   HALVE the default amplitude
0    Use varaiable area fill when plotting
```

(note last line must end with a CR-LF pair.)
Any text editor such as EDLIN can be used to create/modify the PLOT.CFG file.

Sample Session

```
C>PLOT c:\seisdat\line1.dat 11 50 1 0 1 10 18 5 a:line1.gra
```

Here Plot is requested to: load file LINE1.DAT from the sub-directory SEISDAT on drive C:; process record numbers 11 though 50; normalize the data using a scan delay of 5 ms; plot the data using default Inches/second vertical scaling, skipping the first 10 ms, with 18 traces/inch. Then automatically dump its graphics data to a file on the A drive called line1.gra rather than printing it.

Typing:
PLOT ?
will produce a help screen giving a brief description of each of the CLI parameters.

To use PLOT interactively just type PLOT with no parameters. Plot will then begin to ask for each of the parameters above. (The OUTFILE option is available only by using CLI.)
Below is a sample dialog:

```
Enter file name to plot -> line1.dat
Enter starting record number  (1) -> 11
Enter ending record number to plot (32000) -> 50
Do you want auto screen dump? 0=NO/1=YES  (0) -> 1
Enter vertical display size in Inches/Second (8.33333) ->
Plot normal ?  0=OFF  1=ON   (0) -> 1
Enter normalize scan-delay in ms (0) -> 5
Enter starting time of plot in ms  (0) -> 10
Enter trace spacing in traces/inch (20)
(Values above 24 degrade hardcopy) -> 18
```

At this point plotting begins with the same results as in the CLI example, except that output will go to the printer attached to printer port 1. With each screenfull the plot is automatically printed. Note that each parameter has a default value assigned. This value is shown in the parentheses (xx). These default values are assumed if a 'return' is entered to the question. Notice that in respose to the question on Inches/Second, only a 'return' was entered. The default value for the ending trace may seem a bit queer. Since PLOT doesn't know in advance what the last record is, it defaults to an arbitrarily large value.

It is possible to mix CLI and IM. for example

```
PLOT line1.dat 11 50
```

PLOT will answer the first 3 questions from the command line, at that point it becomes interactive, and asks for the rest of the parameters. Again output re-direction is only possible by completing the entire command line.

If a mistake is made during data entry in IM, type CTRL-C to abort the program. Once the program has started plotting it may be aborted by hitting the space bar. If the program is currently printing, hitting the space bar will abort the print and return to fill the screen with more data. Simply hit the space bar a second time to completely stop the program.

If an error occurs during the plot process, it is possible that the program will return to MS-DOS still in graphics mode. If this occurs, run the program RESET. This will return the screen to normal text mode.
i.e.
C>RESET

3.2 TRHD (dump chosen TRace HeaDer words)

Input data: Field / CDP-sorted / CDP-stacked / Others

User's supply: Input filename,
Trace-header word index

Function: To list information contained in the chosen trace-header words (see description of trace header in section 1.2).
Option is also available for hard-copy output.

3.3 TRHDCHNG (TRace HeaDer value CHaNGe)

TRHDCHNG USAGE

```
TRHDCHNG ?  Gives help list
TRHDCHNG  (no parameters) Interactive questions
TRHDCHNG fname
Where:
FNAME  = Input file name to change.
         This file must be in standard KGS-SEG-Y file format.
```

The TRHDCHNG program allows the user to change trace header words in EAVESDROPPER data files. Up to 10 words at a time maybe changed. TRHDCHNG can only do 'universal' changes. It is intended only for those users who are sure of what they are doing. When TRHDCHNG runs, it changes a given header word to a specified value in EVERY trace in the data set. It does not allow for incrementing or decrementing values. Or for skipping some traces and not others. EVERY trace of data is effected.

USAGE:
At the MS-DOS command prompt type

C> TRHDCHNG <cr>

TRHDCHNG will present a message warning of the dangers of improper use and ask for the INPUT FILE NAME. Enter here the name of the file you wish to change. Next you are prompted for the OUTPUT FILE NAME. DO NOT USE THE SAME NAME! TRHDCHNG will output the results to a different file, leaving the original intact. This will allow you to go back to original data should the changes be wrong. After answering this question TRHDCHNG presents the first 96 trace header values. Since many of the words are currently undefined there will be many large or possibly negative values that have no meaning. You may ignore them or use TRHDCHNG to set them to 0. (There is however no reason to do so, since none of the EAVESDROPPER software will use them.)

TRHDCHNG then prompts for the number of words to change. Enter a number from 1 to 10. Enter 0 to exit. TRHDCHNG then prompts the word number to change. This value should be number from 1 to 120. (Refer to section 1.2 for the meaning of the trace header words.) The user is then prompted for the value to insert at this header location. They should be an integer value in the range of -32768 to +32767.
These questions are repeat 'number of words to change' times. TRHDCHNG then presents the list of words to be changed, showing the current value and the new value. The user is ask if it is correct. Answering Y will cause TRHDCHNG to start the changes.

TRHDCHNG may also be started in the following way:

C> TRHDCHNG INPUT.DAT <cr>

In this mode the warning message is skipped, the INPUT DATE FILE is presumed to be INPUT.DAT and the question/answer sessions begins with ENTER OUTPUT FILE NAME.

It is advised that the input file be retain until you are sure that the changed file is what you want. It is also very important to realize that TRHDCHNG changes every single trace of data in the input stream. It changes a specified trace header to a constant value. So to use TRHDCHNG to change the channel number for instance could render the data useless, as all channel numbers would be set to some fixed value.

3.4 BISCONV (Convert Bison Geopro data to KGS format)

Input data : Biscon GeoPro seismograph data files

Output data: KGS data files

Updated trace header words:
1, 2, 6, 8, 15, 16, 55, 58, 59, 88, 92

BISCONV (Bison Conversion) is designed to accept data files generated by the Bison Geo-pro series of field seismographs and convert them to KGS-SEG-Y file format.

Currently Bison instruments uses a communications program (Mirror) to transfer data from its Geo-Pro seismograph to PC disk. During this transfer Mirror creates a file for each shot point of data. The KGS-SEG-Y file format requires that these multiple files be reformatted and merged into one file with appropriate header information.

BISCONV is totally user interactive. The program asks several questions and then converts the files. The names of the input files must have the following structure:

PREFxxx.eee
Where PREF is some prefix name that all files created by Mirror have in common. xxx is a series of ascending numbers i.e. 001 , 002 , 003. eee is an optional extension on all of the file names. A sample directory might look as follows:

REC001.DAT
REC002.DAT
REC003.DAT
REC005.DAT
REC004.DAT

or perhaps

DATA0009
DATA0010
DATA0011
DATA0012

The Bison data sets MUST have this type of naming structure for BISCONV to work. The exact names are not important, only that they all begin with the same prefix, and have ascending numbers in their names.

To invoke BISCONV type BISCONV at the MS-DOS prompt and press return. The program then displays the following opening message.

WELCOME OPERATOR TO THE BISON->KGS PRE-PROCESSOR

This program converts the Bison data files created by Mirrors into to KGS SEG-Y file format.

Program assumes there are 1 or more files to convert with similar names containing ASCENDING numbers, i.e. REC001.DAT, REC002.DAT ... REC099.DAT .

Press RETURN when ready to proceed, CTRL C to abort

At this point the operator may abort the program by typing CRTL-C. Otherwise pressing a return begins the question/answer session. The first question is:

Enter the PREFIX name of the files, (i.e. REC) ->

Enter the prefix used on all of the data files. From the above example directories it would be REC or DATA. Then press the return.

The next questions is about the optional extension

Please enter the EXTENSION of the files if present. i.e. .DAT
Include the . (PERIOD) in name! ->

The files may or may not have extensions. If they do be sure to enter the leading . (period) in the name. The program will check to make sure its present. If there is no extension on the file names then just hit return.

The next questions ask what file numbers to be used. Remember that the files to be converted must have a set of consecutive ascending numbers.

Next enter the file numbers for the records to be changed. For instance, if there are to be 3 files converted named REC014, REC015 , AND REC016, the first number you should enter is 014 , then when asked, enter 016.

REMEMBER TO ENTER LEADING 0's i.e. 014 not 14

Please enter STARTING number, ENTER LEADING 0's! ->

Here one might enter 001 or 0011 as from the example directories above. IT IS VITAL that any leading zeros be entered. If the first file to be converted is called DATA0002, then you must enter 0002 to this question NOT 2, or 02 or 002!

The next question asks for the ending number.

Please enter ENDING number, ENTER LEADING 0's! ->

Again you must enter any leading zeros that are present. The program has a number of checks and will examine your input to make sure it is consistent.
The program then shows the files selected and asks if it is correct. For the example directories above it might read:

You have chosen files :

REC001 through REC005

Is this correct? (y/n) ->

Answer this question with a n followed by a return if you have made a mistake.

BISCONV next asks for the output file name. This is the KGS-SEG-Y file that will be created by the program.

Please enter the name of the output file ->

To this enter any valid MS-DOS file name, including path if desired. i.e. LINE1.DAT , C:\SEIS\LINE1.DAT .

Finally BISCONV asks for the KGS starting Source Sequence Number (SSN).

Please enter the STARTING kgs Source Sequence Number (SSN) ->

This can be any number the operator chooses. Normally it is the same as the first input file number.

Bisconv next checks to insure that the data files are present and opens its output file. This output file is opened in APPEND mode. This means that if the name given for output is already present, it will append new data to the bottom of it. This can be used to combine sets of Bison data that have different prefixes or missing files in the numbering sequence. By running BISCONV more than once with different input file names/ranges and using the same output file name, one can concatenate data sets. If the output file does not exists it is created.

The program then computes how many bytes of disk space will be needed for the operation. It informs the operator of this and also computes how many bytes will be free on the disk after the operation. The operator is then asked if the input files should be deleted as they are converted. On large hard disks this probably will not be necessary. On floppy disks however it is quite possible that there will not be enough room for both the original Bison files and the KGS-SEG-Y file. By selecting the auto-delete function there should be enough room to convert the data on all but the most crowded of disks.

After answering the auto-delete question BISCONV begins the conversion. Conversion is very rapid taking about 2.5 sec / shot on a hard disk system. When BISCONV has completed the conversion it returns the operator to the MS-DOS prompt. The operator can then begin the processing of the data.

If during the processing an error occurs , i.e. disk full or input file not found, BISCONV reports an error and aborts the task, saving what ever it has done up to that point.

3.5 VELP (VELocity Picking from reflection events)

Input data: Field / CDP-sorted data

User's supply:

1. Input filename,
2. If input data is field data, enter Source Sequence Number. If input data is CDP-sorted, enter CDP number.
3. Starting time (in ms) to display data.

PROGRAM DESCRIPTION

The program computes average velocity, two-way vertical travel-time, and depth of reflector from single field record or CDP record. The computation is based on a least-square hyperbolic curve-fitting method using relationship between reflection times and shot-to-receiver distance.

This program graphically displays record data corresponding to the requested record number (either source sequence number or CDP number). It then allows user to simply move a screen cursor (use Arrow Key) to select points along the chosen reflector, and press a key (F1) to record values of time and shot-to-receiver distance. After 2 or more points are selected, the average velocity, zero-offset (two-way) time, and depth can be calculated and printed at the bottom right corner of the screen.

3.6 VSCN (Velocity SCaNning analysis)

Input data: CDP-sorted data

Output data: NMO-corrected data either as
CDP stacked or as CDP gathers

User's supply:

Input CDP-data filename,
Output VSCN-data filename,

Starting & ending CDP-numbers for first CDP-group,
(If need more than 1 CDP-group, enter CDP-group increment)

Record length (ms) to process,
Sample stretch ratio [default = 0.5],

Option to stack CDP,
Trial velocities

PROGRAM DESCRIPTION

The program applies NMO corrections with user's entered trial velocities to the same panel of CDP data.

After the program run is finished, the output NMO-corrected data can be displayed by using "PLOT" .

4. CARD-CONTROL PROCESSING (Version 2.0)

Card-control processing allows the flexibility of applying various procedures in any order. In selected cases the user may be able to go directly from field data to brute stack in a single processing run, while saving all intermediate processed data sets.

Available card-control processing procedures are:

- trace sorting and resorting,
- trace editing,
- static correction (including elevation, automatic, & residual),
- normal moveout correction,
- CDP stacking,
- AGC (Automatic Gain Control) scaling, and
- filtering.

4.1 DIFFERENCES FROM VERSION 1.x

Improvement for flexible usage that makes Version 2.0 of card-control processing program different from the previous Versions (1.x) is summarized as follows.

1. Unlimited size of processing parameters (within the limit of hardware and operating system restriction).

   Version 2.0 employs dynamic memory management (which allocates program memory at run time), as opposed to static memory management in Version 1.x (which fixes program memory at compile time). Therefore under version 2.0 there is no restriction of user's input parameters (within the limit of hardware and operation system).

   For example Version 1.x restricts
   - Size of input seismic data to be up to 1000 samples per trace and up to 24 recording channels.
   - Surface static correction (*SURF) allows maximum of 200 stations to be defined.
   - Trace sorting (*SORT) accepts up to total of 4800 good traces, 10 source-receiver patterns, 3 survey lines (or repeated shots), and 128 folds for secondary sort.
   - etc.

   In Version 2.0 these restrictions have been removed.

2. Same processing operation can be applied more than once within a single run.

   Within one pass user can choose to apply more than one (if appropriate) processing operations with different values of processing parameters.

   Sample carddeck below shows usage of two filtering operations: pre-stacked (with 100-400 Hz bandpass) and post-stacked (with 150-350 Hz bandpass).

```
>>Start
  *Inpf     field.dat
! Pre-stacked filtering
  *Filt     100  400
  *Nmot
  Velf 200  20   350
  *Stak
! Post-stacked filtering
  *Filt     150  350
  *Scal     50
  *Outf     stack.dat
>>End
```

Note that Version 1.x also accepts the same operation more than once within a single pass. But values of processing parameters will be set according to the last set of control-cards for that operation. (For example, using the above deck, Version 1.x will apply both pre-stacked and post-stacked filtering with 150-350 Hz bandpass according to the last *Filt-card, which is incorrect for the first filtering operation.) Version 2.0 has corrected the problem.

3. Processing history handling.

In order to keep track of processing history, the program automatically generates a text file (containing all previous processing parameters) for each output seismic data set. Filename of processing history is stored in the trace header of a corresponding seismic data set. At the time of plotting, the plot program will also print these processing parameters along with the plot of seismic data.

Please refer to section on "Processing history handling" for further explanation.

4. Changes in card format.

The following cards have been changed in order to be more flexible for general usage.

>>START (add option to keep all created temporary data set)
PN (add parameters to define source-receiver pattern)
*SCAL (add option for root-mean-square AGC scaling)
*STAK (add option to use square of CDP-folds)
*STAT (add option to apply bulk statics)

For further explanation please refer to section on "Control-card Format Description: Processing-Operation Cards" for PTRN (under *SORT), *SCAL, *STAK, and *STAT, respectively.

5. Remove restriction on trace sorting (*SORT).

Field geometry assignment & trace sorting operation (*SORT) can be applied to any type of seismic data set. Version 1.x restricts *SORT operation to accept only raw field data set. Version 2.0 has removed this restriction.

6. Additional processing operations.

Additional processing operations are
1) Automatic trace editing (*AUED)
2) Filtering (*FILT).

Please refer to section on "Control-card Format Description: Processing-Procedure Cards" for *AUED and *FILT, respectively.

4.2 HOW TO RUN THE PROGRAM

To run the program, type

```
SEIS  LIST.INP  LIST.OUT
```

where
SEIS is the command for executing the program run,
LIST.INP is the control-deck file (test file prepared by user) containing processing parameters, and
LIST.OUT is the output list-file to be generated during the run, which contains the processing history of the run.

Alternatively, user can type

```
SEIS
```

with a return key. The program will then ask for the input control-deck and output-list filenames.

After completion, user can either view the generated output-list text file by typing

```
TYPE LIST.OUT
```

with a return key, or get a hard copy by typing

```
PRINT LIST.OUT
```

with a return key.

4.3 HOW TO PREPARE CARDDECK

4.3.1 Card Size

Maximum characters per card = 250 (including blank spaces)
Maximum parameters per card = 40
Maximum characters per parameter = 40

4.3.2 Card Type

Cards are classified as comment cards, marker cards, I/O operation cards, and processing-operation cards.

1) Comment Cards.

Comment card starts with ! as the first character in the line. It is for documentation only. Comment card can appear anywhere in the control-deck file and will be ignored by the processing program.

Only those comment cards between >>START and >>END that will be transferred to output-list file.

2) Marker Cards.

Marker cards start with >> as the FIRST TWO characters.
>>START Card is to mark the beginning of the processing flow.
>>END Card is to mark the end of the processing flow.

3) I/O-Operation Cards

I/O-operation cards start with * as the FIRST character.
*INPF Card is to define input seismic filename to process.
*OUTF Card is to define output processed filename.

4) Processing-Operation Cards

Processing-operation cards consist of
i) Procedure-definition Card, and may be followed by
ii) Procedure-data Card.

Procedure-definition cards start with * as the FIRST character.

Data cards do not have a *-prefix and generally define the processing parameter values used.

Below are examples of a control deck.

```
! This is a comment card
! Blank card is also allowed anywhere
! >>START-card marks the beginning of processing
  >>START

! Assign "SORT.DAT" as input seismic data file
  *INPF    SORT.DAT

! Apply normal-moveout correction
! Here the velocity function at CDP 200 is 235 ft/s or m/s
! (Be consistent with units) at two-way time 20 ms
  *NMOT
  VELF    200      20  235

! Direct output data to "NMOT.DAT"
  *OUTF   NMOT.DAT
! >>END-card marks the end of processing
  >>END
```

The same processing flow would result if all the comment cards are removed.

```
>>Start
*Inpf     SORT.DAT
*Nmot
velf  200   20 235
*Outf     NMOT.DAT
>>End
```

Within each card, parameters are in FLEXIBLE (FREE) format, i.e. they are separated by blank space(s). Blank lines can also optionally be inserted any places.

Keywords are case-insensitive, e.g. *NMOT, *Nmot, or *nmot yields the same effect.

4.3.3 EXAMPLES

This section only provides examples of carddeck organization. Please refer to section on "Control-card Format Description" for detailed description for each card.

1. Generate sorting table

In order to check for correctness of sort-deck, often times it is useful to generate a corresponding sorting table without the actual trace sorting. In this case, only *SORT-operation & data cards are required.

```
>>START
  ! Trace sorting (*SORT) according to CDP number (12),
  ! within each CDP-gather, sort traces in decreasing order (-1)
  ! of source-to-receiver distance (19)
*SORT  12 19  -1

  ! Define source-receiver pattern (PTRN) with 24 recording
  ! channels (default).  Receiver spacing is .25 feet.
  ! (Unit of feet is a default.)
PTRN  .25
  ! ******* Note that PN-card is different from Version 1.x
  ! Define source-receiver Pattern Number (PN) 1 with
  ! source at station # 100
  ! the first receiver at station # 124
  ! there are total of 24 receiver with station # decremented
  ! by 1, i.e. the second receiver at station # 123, ....,
  ! and the 24th receiver at station 101
  ! i.e. this pattern is end-on
PN   1  100    124 24 -1

  ! Geometry Assignment based on Source Sequence Number
SHOT
  ! For the first SN-card,
  ! assign Source Sequence Number (SN) 20
  ! with source at station # 95
  ! using source-receiver pattern # 1
  ! in-line offset of the source = 0 unit length (use feet here)
  ! off-line offset of the source = 0 unit length (feet)
  ! repeated-shot # = 1
  ! omit channel number from 20 to 24
  !
  !  ssn# shot ptrn# in-line off-line rep-shot#  bad_chans
SN    20   95     1      0       0        1         20  24
SN    21   96     1      0       0        1         21  24
SN    22   97     1      0       0        1         22  24
SN    23   98     1      0       0        1         23  24
SN    24   99     1      0       0        1         24  24
SN    25 100
  ! skip ssn#26 to #29
  ! Group assignment
  ! from source sequence numbers 30 to 60, incrementing by 1,
  ! correspondingly assigned to source at station numbers
  ! 101 to 131 incrementing by 1
  ! with defaults of
  ! source-receiver pattern # 1
  ! zero in-line & off-line offset for source
  ! repeated-shot # 1
  ! and do not omit any channel
SNSN  < 30 60 1 >  < 101 131 1 >
  ! Print the table of sources and receivers
TABL

>>END
```

2. <u>Common Channel Sorting with 2 repeated-shots</u>

```
>>START
*INPF  LYNDON.DAT
  ! Trace sorting based on channel number (8)
  ! within each channel, sort traces in increasing order
  ! of source sequence number (92)
*SORT  8  92

  ! Define source-receiver pattern with
  ! 24 recording channels (default)
  ! receiver spacing of 3 feet  (unit of feet is a default)
PTRN   3

  ! ******* Note that PN-card is different from Version 1.x
  ! Define source-receiver pattern number 1
  ! Source at station # 100.  There are 2 sets of receivers
  ! First set: 1st receiver at station # 84
  !            2nd receiver at station # 85 (increment by 1)
  !            ......
  !            last receiver (12th) at station # 95
  ! Second set: 1st receiver at station # 105
  !             2nd receiver at station # 106 (increment by 1)
  !             ......
  !             last receiver (12th) at station # 116
  ! I.e. this pattern is split-spread
PN  1  100    84 12 1     105 12 1

  ! Geometry assignment using source sequence number
SHOT
  ! For this data set, there are 2 records for each shot-station,
  ! i.e. shot (and record) twice at each station.
  ! Sortdeck below will separate data into 2 files: one for
  ! the first shot at each station; another for the second shot.
  ! For the first 2 SN-cards:
  ! assign source sequence # 1 with source at station #100
  !                                 source-receiver pattern #1
  !                                 zero in-line, off-line
  !                                 repeated shot # 1
  ! assign source sequence # 2 with repeated shot # 2 and
  !                                 everything else the same.
SN  1  100  1   0  0  1
SN  2  100  1   0  0  2
SN  3  101  1   0  0  1
SN  4  101  1   0  0  2
SN  5  103  1   0  0  1
SN  6  103  1   0  0  2
SN  7  104  1   0  0  1
SN  8  104  1   0  0  2
  ! Sort data into 2 separated files
*OUTF   LYNSRT1.DAT   LYNSRT2.DAT

>>END
```

3. NMOT & STACK

```
>>START
  !processing only from cdp #200 to #250
*INPF   FCDP.DAT   200  250
*NMOT
VELF  225   10  2000  50  3000
*STAK
*OUTF  FSTAK.DAT
>>END
```

4. RESORT back to common source sequence

```
>>START
*INPF  FNMOT.DAT
  ! Trace resorting into (increasing) order of
  ! source sequence number (92)
  ! also sort traces within each source sequence number
  ! in increasing order (1) of channel number (8)
*RSRT  92  8  1
*OUTF  FRESORT.DAT
>>END
```

5. AUTS

For 2 runs:

```
>>START
*INPF    FNMOT.DAT
*AUTS     XAUTS.TBL   3   20   11  3  0.7
GCEN    200    50    280   55
>>END

>>START
*INPF   FNMOT.DAT
  ! **** Note that *STAT-card is different from Version 1.x
*STAT   2  XAUTS.TBL
*OUTF   FAUTS.DAT
>>END
```

Or for a single run:

```
>>START
*INPF    FNMOT.DAT
*AUTS    XAUTS.TBL  3  20  11  3  0.7
GCEN    200 50  280 55
  ! **** Note that *STAT-card is different from Version 1.x
*STAT   2  XAUTS.TBL
*STAK
*OUTF   FAUTS.DAT
>>END
```

4.4 PROCESSING-HISTORY HANDLING

The purpose is to help user to keep track of processing history of the processed seismic data set. For each output seismic data set, card-control program will automatically generate a corresponding processing-history text file. Therefore each output seismic data set processed by card-control program will have a unique processing-history file associated with it, within the same directory.

Processing-history filename will be stored in the trace header of seismic data. When plotting the seismic data, the plot program will look for processing-history filename in the trace header and automatically display the content in a history file along with the plot of seismic data.

4.4.1 Processing-History Filename

Processing-history filename will be automatically selected by card-control program. It has the same primary name as its seismic data filename. However, its extension name starts with H and followed by 2 digits.

For example, seismic filename

MACSORT.DAT

will have the corresponding history filename

MACSORT.Hxx

where xx is any 2-digit number.

The program will start the 2-digit number with 00. So, for example, if file MACSORT.H00 does not exist within the directory of the output seismic data, the selected history filename will be MACSORT.H00. However, if it does not exist, the history filename will be MACSORT.H01 (if MACSORT.H01 does not previously exist), i.e. the 2-digit number will be incrementing; 00, 01, 02, 03, output seismic data. (This allows the maximum of 100 filenames, H00- H99, having the same primary name.)

Therefore with the above strategy, history filename will be unique, even if there are more than one output seismic data sets with the same primary name (e.g. MACSORT, MACSORT.DAT, MACSORT.SRT, MACSORT.M).

4.4.2 Content of Processing History

Processing-history file contains all control-cards that have been applied to the corresponding seismic data.

For example, below is a carddeck of the *SORT operation applied to raw field data. Raw field data (RAWFIELD.DAT) has no history file associated with it.

```
!CDP sortdeck
>>Start
*Inpf      RAWFIELD.DAT
*Sort
Ptrn .25
Pn 1 100 101 24 1
Shot
Snsn < 10 20 1 > < 100 110 1 >
*Outf      CDPSORT.DAT
>>End
```

Output sorted seismic data (CDPSORT.DAT) will have a corresponding history file (CDPSORT.H00) with content

```
!Processing history file of CDPSORT.DAT
*Inpf      RAWFIELD.DAT
*Sort
Ptrn .25
Pn 1 100 101 24 1
Shot
Snsn < 10 20 1 > < 100 110 1 >
```

Now suppose that *NMOT and *STAK operations are applied to seismic data set CDPSORT.DAT. The carddeck is

```
! NMO & Stack deck
>>Start
*Inpf    CDPSORT.DAT
*Nmot
Velf  200   20  335
*Outf    NMOT.DAT
*Stak
*Filt   150  350
*Scal   50
*Outf    STACK.DAT
>>End
```

The output NMO-correct seismic data set (NMOT.DAT) will have a corresponding history file (NMOT.H00) with content

```
!Processing history file of NMOT.DAT
*Inpf      RAWFIELD.DAT
*Sort
Ptrn .25
Pn 1 100 101 24 1
Shot
Snsn < 10 20 1 > < 100 110 1 >
*Nmot
Velf  200   20  335
```

And the output stacked seismic data set (STACK.DAT) will have a corresponding history file (STACK.H00) with content

```
!Processing history file of STACK.DAT
*Inpf      RAWFIELD.DAT
*Sort
Ptrn .25
Pn 1 100 101 24 1
Shot
Snsn < 10 20 1 > < 100 110 1 >
*Nmot
Velf  200   20  335
*Stak
*Filt   150  350
*Scal   50
```

Therefore each output seismic data set from the card-control program will have a complete history of control-cards beginning at the first input seismic data (in this case is raw field data) and ending at the last control-card prior to the output processed data.

4.4.3 How the Program Handles

At the beginning of the run, card-control program will generate a temporary working file (in the working directory)

HISTORY.Bxx

(where xx is 2 digits, i.e. the working file will be HISTORY.B00 if it does not previously exist, else the digits will be 01, 02,

If a history file of input seismic data exists, the program will copy its content to the working file (HISTORY.Bxx). Otherwise, the working file is kept empty.

As each control-card is read from the input deck, it will be appended into the working file. However, the *OUTF-card will not be appended. Instead, when the program found that the card is *OUTF, it will create a corresponding history file and copy the current content of the working file (HISTORY.Bxx) into the created history file.

At the end of the program run, the temporary working file (HISTORY.Bxx) will be automatically deleted.

4.4.4 Exception Handling

Exception cases may occur as follows.

1) Rerun the same carddeck (with no change in *OUTF-card).

Often times user wants to rerun the same carddeck with minor changes in processing parameters. Assuming here that there is no change in *OUTF-card.

In this case the program will refresh the content (destroy the old content) of both output seismic data and its history file.

Therefore whenever the *OUTF-card is read, the program will first check if the history file of the output seismic data (defined in *OUTF-card) exists or not. If it does exist, the program will delete it first (before generate another unique one).

In other words, the card-control program takes care of clean-house. (If not, disk space will soon be occupied with the unused history file, since the program always tries to generate a unique history file. Also this would be frustrated for the user to manually do the clean-house himself.)

2) User aborts the run.

There are times when user decides to abort the program while it is running. In this case the temporary working file (HISTORY.Bxx) will be left in the disk. User has to take care of deleting it himself. If not deleting it, the program will generate a unique working file (with different digits in the extension name) in the next run.

In other word, whenever the card-control program is NOT running, and file(s)

HISTORY.Bxx

exists in the working directory, user can delete it in order to clean disk space.

(Note: The reason of having the card-control program keep on generating a unique working file lies in the correct result even when the program is run concurrently in the multi-tasking mode of, for example, OS/2 operating system.)

3) Lost history file.

History file may be deleted accidentally (or intentionally). When the program cannot find the history file of the input seismic data set, it will give a warning message and proceed the processing. History file will be constructed based on the available processing parameters.

4.5 PROCESSING FLOW

There are 3 main steps in processing flow:

1) Read carddeck.
   - Allocate memory for object module to handle the processing-operations that are defined in the carddeck.
   - Prepare processing flow (briefly explained below).

2) Process data.
   Within this step, there are 3 parts:
   2.1) Pre-processing: Allocate working memory for each processing-operation.
   2.2) Processing: Process seismic data trace by trace. Each trace is processed operation by operation.
   2.3) Post-processing: Deallocate working memory.

3) Deallocate memory of object-modules (that handle processing-operations) and exit the program.

During the program run, only steps 2.1 (Pre-processing) and 2.2 (Processing) are visible to user.

In pre-processing step, the program lists the user's defined processing operations (on the CRT).

In processing step, the program report progress by printing the current processed record number (on the CRT).

Therefore card-control program processes seismic data sequentially trace by trace (step 2.2). There are exceptional cases (*AUTS, *RESE, *SORT, *RSRT), however, where processing data set by data set is needed. In those cases, the program will split processing flow into phases. The program will process the data trace by trace only within each phase. At the end of each phase, the program has to complete processing of the whole seismic data set before proceeding to the next phase.

Usually when processing flow is split into phases, temporary output disk files (for intermediate processed seismic data) are automatically created. These working diskfiles will be automatically deleted at the end of the program run (unless a flag is set in >>START-card; see explanation of >>START-card).

The breaking up of processing flow into phases is designed to handle variety of carddeck organization (offering flexibility to user) and to speed up the processing time.

4.6 PROGRAM MEMORY AND SPEED

As a result of employing dynamic memory management, Version 2.0 of card-control program requires the overhead in managing the memory dynamically.

In term of speed, the overhead mainly involves multiple readings of carddeck (e.g. scanning through carddeck to find total number of user-defined parameters, allocating memory space to hold those parameters, and finally re-read the carddeck to actually store the parameter values). This may slightly effect the program speed.

Program memory for Version 2.0, before any memory allocation for user-defined processing operations, is approximately 185 Kbytes. (Compare with Version 1.x which has static program memory of over 512 Kbytes.) Therefore program memory left for user-defined processing operations in each run is approximately 400 Kbytes under MSDOS (which is considered to be plenty).

4.7 ERROR HANDLING

There are 2 levels in error handling.

1) Warning level. For this level the program will give a warning message and proceed the execution.
2) Error level. For this level the program will give an error message and abort the execution.

Generally, causes of errors are

1) Invalid/wrong card format.
   Some of pre-processing error messages are listed below. Please refer to "Control-card Description" section for error messages of each control-card.

   ```
   Missing >>START-card
   Missing >>END-card
   No processing card
   No *Inpf has been specified
   Line too long ( > 40 words)
   Word too long ( > 40 characters)
   Invalid card found
   ```

2) Run out of disk space.
   The program also checks disk space while writing the processed seismic data to disk. If the disk is full, the program will give a disk-full message and abort itself.

3) Run out of program memory.
   This should rarely occur. However, the program always check for sufficient memory before allocating it. If the memory is not enough, the message

   Insufficient memory for dynamic allocation!

   will appear and the program will abort itself. User should either split carddeck into several passes, or reduce the size of parameters (e.g. in data-card).

4.8 CONTROL-CARD DESCRIPTION

This section describes the format of control-card for each processing operation.

Control-card format is tabulated into 3 columns (from left to right of the reader): Field Number, Code or Parameters, and Default Value.

Field Number is only for identification of Code or Parameter. Some Field Numbers are enclosed with Square Bracket. This implies to use the Default Value (appeared in the third or most-right column) if not supplied by the user.

The Default Value for a particular Field is indicated in the third column of the table.

User must be responsible to supply Codes or Parameters for those Fields that do not have the Default Value. These are Fields with indicated "NA" (for Not Available) in the third column of the table.

Most required Parameters, if not explicitly indicated, are in INTEGER 16-bit fixed point format, which ranges from -32768 to +32767. Some Parameters can be FLOATING POINT, which will be explicitly indicated.

4.8.1 MARKER CARDS

>>START (mark the START of card-control processing)

>>START Card

Field	Code / Parameter	Default
0	**>>START** - Keyword to mark the beginning of processing procedures	NA (Not Available)
[1]	Flag to save/erase temporary output data at the end of processing (if temporary data are created) 0 = erase 1 = save	0

Note: Temporary output data files will be automatically created when they are needed (usually when processing flow is split into several phases).

General temporary output-filename is

XXn.Tmm

where n is some number and mm is a 2-digit number. (Primary name starts with XX, and extension name starts with T).

Temporary stacked and pilot data filenames are

XXSTACK.Tmm

and

XXPILOT.Tmm,

respectively. (These are generated in *AUTS and *RESE operations.)

Processing operations that may cause temporary data (or working disk space) to be created are

*AUTS (Generate automatic statics table)
*RESE (Residual statics)
*RSRT (Trace re-sorting)
*SORT (Trace sorting).

At the end of the run, the program will either erase or save the created temporary working files, depending on the flag in Field #1 of >>Start-card. Default value is 0 (erase all created temporary files).

>>END (mark the END of card-control processing)

>>END Card

Field	Code / Parameter	Default
0	**>>END** - keyword to mark the end of processing procedures.	NA
None		

4.8.2 I / O OPERATION CARDS

*INPF (INPut seismic data File)

***INPF Card**

Field	Code / Parameter	Default
0	***INPF** - Procedure Definition Code	NA
1	Input data filename (String of alphanumeric characters)	NA
[2]	First record number to process * = if not specified, use first record in input data file.	*
[3]	Last record number to process ** = If not specified, use last record in input data file. If specified, be sure to specify First record number (FIELD 1). Otherwise it will itself become the First record.	**
[4]	Record length in ms (milliseconds) to process (can be FLOATING POINT) *** = If not specified, use total record length in input data file. If specified, be sure to specify First record (FIELD 1) and Last record (FIELD 2).	***
8Q None		

NOTE: 1. For FIELD 1, current MSDOS allows a maximum of 8 characters for filename with a maximum of 3 characters for extension, e.g. LYNDON.DAT, F2345678.TMP, E:S2345678.LST (where "E:" indicates disc drive "E").
2. If parameters in Field 2, 3, or 4 are provided, processing will be ignored out of the specified range.
3. For FIELDs 2 and 3,
If input data is raw field data, will use source sequence number (header word 92).
If it is CDP data, will use CDP number (header word 12).
If input data is neither raw field nor CDP type, will use Record Number Index from value in trace header word 3.

*INPF Error Messages

1) Input file ... does not exist
2) Input file ... has invalid record-index (header word#1 = ..)
3) Cannot find last record in input file ...
4) Input file ... has First Record# (..) > Last Record# (..)
5) Starting record# .. is not in input seismic data
6) User's specified first record# (..) > Last record# (..) in seismic data
7) Specified last record# (..) < First record# (..) in seismic data
8) Processing First record# (..) > Processing Last record# (..)
9) Allow only 1 *Inpf-card per pass;
10) Field# 1 is missing

*INPF Warning Messages

1) Record# ... is missing in input seismic data set
   Processing will begin at record# ....
2) Specified first record# (..) < First record# (..)
   in seismic data
   Reset specified first record# to be ..
3) Specified last record# (..) > Last record# (..)
   in seismic data
   Reset specified last record# to be ..
4) Specified record length (.. msec) > Input data (.. msec)
   Reset specified record length to be .. msec

*OUTF (OUTput seismic data File)

*OUTF Card

Field	Code / Parameter	Default
0	***OUTF** - Procedure Definition Code	NA
1	Output-data filename (String of alphanumeric characters)	NA
[2]	Second output-data filename (if needed)	
.	.	
.	.	
	NOTE: 2 (or more) output files are allowed ONLY for CDP SORTING with 2 (or more) repeat-shots.	

NOTE: Updated header word : 58

"*OUTF" Error Messages

1) Missing output filename

2) Cannot open output file ....
   --- You may run out of disk space.

3) Need .. more outf-filenames to match # repeatshots
   --- i.e. number of filenames in *OUTF-card is less than number of repeated-shots defined in field-geometry assignment (via *SORT operation).
   This message occurs only during trace-sorting and field-geometry assignment (*SORT) operation. When there are more than one repeated-shots, number of output sorted files must match number of repeated-shots.

4) # outf-filenames exceeds # repeatshots
   --- i.e. number of filenames in *OUTF-card is greater than number of repeated shots defined in field-geometry assignment (via *SORT-operation).
   This message occurs only during trace-sorting and field-geometry assignment (*SORT) operation. When there are more than one repeated-shots, number of output sorted files must match number of repeated-shots.

5) While output to file ....
   Run out of disk space @Record# .. Trace# ..
   --- Run out of (physical) mass-storage disk space while outputting seismic data (during processing).

4.8.3 PROCESSING OPERATION CARDS

*AUED (AUtomatic trace EDiting)

Procedure Definition Card: "*AUED"

"*AUED" PROCEDURE DEFINITION CARD

Field	Code / Parameter	Default
0	***AUED** - Procedure Definition Code	NA
1	Noise window size (in ms)	NA
[2]	Acceptable signal-to-noise ratio (FLOATING POINT)	1.0
[3]	Flag to list signal-to-noise ratio 0 = Do not list 1 = List	1
[4]	Flag to list bad traces 0 = Do not list 1 = List	1
[5]	Flag to omit bad traces from output data file 0 = Do not omit 1 = Omit	0
None		

"*AUED" PROCEDURE DESCRIPTION

Automatic-trace-editing operation calculates signal-to-noise ratio based on user-defined noise window (Field # 1), where noise window is from 0 to the value in ms specified by user.

Signal-to-noise ratio is defined here as

$$S / N$$

where S is the root-mean-square amplitude for the whole trace, and N is the root-mean-square amplitude within the user-defined noise window. Or

$$S = \left(\sum_{i=1}^{M} (A[i])^2 \right)^{1/2} / M$$

and

$$N = \left(\sum_{i=1}^{W} (A[i])^2 \right)^{1/2} / W$$

where A[i] = seismic amplitude at sample i^{th},
M = total samples in the seismic trace, and
W = number of samples in the user-defined noise window.

With this approach, traces with large amplitude in the noise window will tend to have small signal-to-noise ratio.

The program will determine whether the trace is good or bad by computing signal-to-noise ratio, and comparing it with the threshold (defined in Field # 2). Bad trace has lower signal-to-noise ratio than the threshold. And good trace has signal-to-noise ratio higher than or equal to the threshold.

For an exceptional case where N is zero, ie. all amplitudes are zero within the noise window, the program will automatically extend the noise window until the first non-zero amplitude is found.

Options are available for listing signal-to-noise ratio (Field # 3), for listing bad traces (Field # 4), and for omitting bad traces from the output data (Field # 5).

*AUTS (generate AUTomatic Statics table)

Procedure Definition Card: "*AUTS"
Procedure Data Card: "PERC" [Optional, 1 Card]
"GCEN" <Required, at lease 1 Card>

"*AUTS" PROCEDURE DEFINITION CARD

Field	Code / Parameter	Default
0	***AUTS** - Procedure Definition Code	NA
1	Output AUTS-table FILENAME (String of alphanumeric characters)	NA
2	Static Limit in ms	NA
3	Gate Size in ms	NA
[4]	Number of summing CDPs to form pilot trace for cross-correlation	3
[5]	Number of static iterations	1
[6]	Cross-correlation coefficient threshold (FLOATING POINT) Note: Must be between 0. and 1. It is the minimum "matching" scale for determining trace shift. 0.0 = poorly match with pilot trace 1.0 = perfectly match with pilot trace Default = 0.5 (mildly match).	0.5
[7]	MINIMUM offset to accept in forming pilot traces (FLOATING POINT)	0
[8]	MAXIMUM offset to accept in forming pilot traces (FLOATING POINT)	32000
None		

"PERC" PROCEDURE DATA CARD (OPTIONAL card to follow *AUTS-Card)

Field	Code / Parameters	Default
0	**PERC** - PERCentage Procedure Data Code	NA
1	Percentage of first CDP forming pilot trace	NA
2	Percentage of Second CDP forming pilot trace	NA
.	.	.
.	.	.

NOTE: "PERC" card indicates weights in summing CDPs to form pilot trace. If this card is absent, will apply equal weight to all CDPs.
If present, user MUST specified number of parameters in PERC-Card as FIELD 4 of *AUTS card. For example, let FIELD 4 of *AUTS card = 5 (use 5 CDPs to form pilot trace), then there must be 5 percentage values in PERC-Card, e.g.

PERC 50 70 100 80 60

will apply weights of
100% to the CDP of interest,
50% and 70% to the lower two CDPs, respectively,
80% and 60% to the higher two CDPs, respectively.

"GCEN" PROCEDURE DATA CARDs (Card[s] to follow *AUTS-Card)

Field	Code / Parameter	Default
0	**GCEN** - Gate-CENter-time Procedure Data Code	NA
1	CDP number for gate center time in FIELD 2	NA
2	Gate center time in ms for preceding CDP in FIELD 1	NA
[3]	CDP number for gate center time in FIELD 4	
[4]	Gate center time in ms for preceding CDP in FIELD 3	
.	.	
.	.	

NOTE: For more than one CDP-Time pair, interpolation of gate center time BETWEEN specified CDPs. For example,
GCEN 200 30 300 50
gate-center-time for CDP #200 and lower is 30 ms;
for CDP #300 and higher is 50 ms;
and will be interpolated in between CDP#201 and #299.

*AUTS & GCEN Error Messages

1. Missing GCEN-card for *AUTS operation

2. GCEN: CDP# (Field# ..) must increase

3 GCEN: GateCenter - HalfGateSize - StaticLimit < 0

4. GCEN: GateCenter + HalfGateSize + StaticLimit > .. ms

5. GCEN: No CDP-time pair

6. Cannot open output Auts-table file ...

7. No good stacked trace has been found.
   Check *AUTS-card for minimum & maximum offsets (Field# 7 & 8)
   or check input seismic data

8. Stacked file exhausted

PERC Error Message

1. PERC: # of defined percentages (..) not match # of pilot CDPs (..)

2. PERC: Normalized weight is zero

*AUTS Warning Messages

1. Field# 4: Change CDPpilots to odd number
   --- if user-defined CDPpilots is an even number.

2. Field# 6: Change threshold to 0.5 (default)
   --- if user-defined threshold is out of range.

"*AUTS" PROCEDURE DESCRIPTION

AUTS Output Table

"*AUTS" procedure computes a set of "Surface Consistent" shot and receiver statics. It outputs a table of shot and receiver static to output filename specified in FIELD 1 of *AUTS card. (With appropriate use of "*STAT" card, user can either apply this static table to correct the data right after *AUTS-procedure within the same run, or he may choose to apply it in a later run.)

For correction of static within the same run right after generating AUTS-table, card deck may look like

```
>>START
*INPF   FNMOT.DAT
*AUTS     FAUTS.TBL  3   30 11 3 0.5
GCEN    200  50
*STAT   2  FAUTS.TBL
*STAK
*OUTF   FAUST.DAT
>>END
```

Or for two runs, giving the same result as above, card deck may look like

```
>>START
*INPF   FNMOT.DAT
*AUTS     FAUTS.TBL  3   30  11  3   0.5
GCEN    200  50
>>END
```

and

```
>>START
*INPF   FNMOT.DAT
*STAT  2      FAUTS.TBL
*STAK
*OUTF   FAUST.DAT
>>END
```

Averaging Technique

"*AUTS" procedure in this system is based on an iterative averaging technique to develop a set of statics for each source sequence number and each receiver station number along the seismic survey line.

With this technique, the input data for "*AUTS" procedure has to be NMO-corrected prior to *AUTS-procedure. The approach is to calculate residual "raw static" shift of each trace with respect to the reference "pilot" trace. This raw static, in turn, belongs to a unique pair of shot sequence number and receiver station number producing a corresponding trace.

From the set of raw statics, surface consistent residual static for each source sequence can be estimated by averaging the receivers' statics iteratively along the common shot plane. And receiver residual static can also be estimated in the same manner along the common receiver plane.

Pilot Trace & Cross-correlation

Pilot trace is formed by summing the surrounding CDPs. To form pilot trace, user can select Number of CDPs (FIELD 4, *AUTS-Card) forming, acceptable Minimum Offset, and acceptable Maximum Offset (FIELD 7 and 8, *AUTS-Card, respectively).

Cross-correlation coefficient (FIELD 6, *AUTS-Card) can be between 0 (poorly match) and +1 (perfectly match). The default value is 0.5. The program will also report number of bad cross-correlation of which the resulting coefficient is lower than this threshold value.

Algorithm

The procedure first stacks the CDP NMO-corrected data (stored in disk file called "XXSTACKxx.TMP", under the same directory of the program run), where xx is a 2-digit number.

Based on this CDP-stacked file, it then forms the pilot trace file called "XXPILOTxx.TMP", where xx is a 2-digit number.

Next, each pilot trace is used as a reference to determine the "raw static" shift by cross-correlated with each input NMO-corrected trace. This process generates a table set of raw static for the whole data set.

After all cross-correlations are done, the program iteratively refines the residual shot and receiver statics by the averaging technique.

Processing Speed

Major step that mainly consumes processing time is cross-correlation. Therefore, large window size (FIELD 3, *AUTS-Card) dramatically slows down the processing. Note that higher number of static iterations only effect processing speed very slightly.

*EDFM (EDit - First arrival Mute)

Procedure Definition Card: "*EDFM"
Procedure Data Card : "TAPR" [Optional, 1 Card]
"FARM" <Required, at least 1 card>

"*EDFM" PROCEDURE DEFINITION CARD

Field	Code / Parameter	Default
0	***EDFM** - Procedure Definition Code	NA
[1]	Trace Header Index for RECORD Number	92
[2]	Trace Header Index for TRACE Number	8
None		

"TAPR" PROCEDURE DATA CARD (OPTIONAL Card to follow *EDFM-Card)

Field	Code / Parameter	Default
0	**TAPR** - TAPeRing Procedure Data Code	NA
1	Taper Length in milli-seconds	NA
None		

NOTE: If "TAPR"-Card is absent, data will be abruptly muted, i.e. with no tapering effect.

"FARM" PROCEDURE DATA CARD (Card[s] to follow *EDFM-card)

Field	Code / Parameter	Default
0	**FARM** - First-ARrival-Mute Procedure Data Code	NA
1	RECORD Number	NA
2	TRACE Number	NA
3	Mute END Time (ms)	NA
[4]	TRACE Number	
[5]	Mute END Time (ms)	
.	.	
.	.	

NOTE: No mute is applied when Mute-END-Time = 0

"*EDFM" ERROR MESSAGES

1. *EDFM: Missing FARM-card
2. FARM: Record (Field# 1) must increase
3. FARM: Trace (Field# ..) must increase

"*EDFM" PROCEDURE DESCRIPTION

First Arrival Muting "*EDFM"-Procedure has the same application as Surgical Muting "*EDMT"-Procedure EXCEPT that Mute Starting Time in "*EDFM"-Procedure is automatically set to zero.

*EDKL (EDit - KiLl Traces)

Procedure Definition Card: "*EDKL"
Procedure Data Card : "KILL" <Required, at least 1 card>

"*EDKL" PROCEDURE DEFINITION CARD

Field	Code / Parameter	Default
0	***EDKL** - Procedure Definition Code	NA
[1]	Trace Header Index for RECORD Number	92
[2]	Trace Header Index for TRACE Number	8
None		

NOTE: Updated header word: 15

"KILL" PROCEDURE DATA CARD (Card[s] to follow *EDKL-Card)

Field	Code / Parameter	Default
0	**KILL** - KILL-trace Procedure Data Code	NA
1	Starting RECORD Number	NA
2	Ending RECORD Number	NA
3	Starting TRACE Number	NA
4	Ending TRACE Number	NA
[5]	Starting TRACE Number	
[6]	Ending TRACE Number	
.	.	
.	.	

"*EDKL" ERROR MESSAGES

1. *EDKL: Missing KILL-card
2. KILL: Missing "ending trace number"
3. KILL: Starting Record # > Ending Record #
4. KILL: Starting Trace # > Ending Trace #

"*EDKL" PROCEDURE DESCRIPTION

Kill-Trace "*EDKL"-Procedure resets flag in trace header word to be dead-trace flag (i.e. reset Trace Header Word #15 to be zero). Dead traces are completely omitted in the output seismic data set.

*EDMT (EDit - surgical MUTE)

```
Procedure Definition Card: "*EDMT"
Procedure Data Card      : "TAPR"   [Optional, 1 Card]
                           "MUTE"   &lt;Required, at least 1 Card&gt;
```

"*EDMT" PROCEDURE DEFINITION CARD

Field	Code / Parameter	Default
0	***EDMT** - Procedure Definition Code	NA
[1]	Trace Header Index for RECORD Number	92
[2]	Trace Header Index for TRACE Number	8
None		

"TAPR" PROCEDURE DATA CARD (OPTIONAL Card to follow *EDMT-Card)

Field	Code / Parameter	Default
0	**TAPR** - TAPeRing Procedure Data Code	NA
1	Taper Length in milli-seconds	NA
None		

NOTE: If "TAPR"-Card is absent, data will be abruptly muted, i.e. with no tapering effect.

"MUTE" PROCEDURE DATA CARD (Card[s] to follow *EDMT-Card)

Field	Code / Parameter	Default
0	**MUTE** - MUTE Procedure Data Code	NA
1	RECORD Number	NA
2	TRACE Number	NA
3	Mute START Time (ms)	NA
4	Mute END Time (ms)	NA
[5]	TRACE Number	
[6]	Mute START Time (ms)	
[7]	Mute END Time (ms)	
.	.	
.	.	
.	.	

NOTE: No mute is applied when Mute-START-Time = Mute-END-Time.

"*EDMT" ERROR MESSAGES

1. *EDMT: Missing MUTE-card
2. MUTE: Record (Field# 1) must increase
3. MUTE: Trace (Field# ..) must increase
4. MUTE: Start time (Field# .. = ..) > End time (Field# .. = ..)

"*EDMT" PROCEDURE DESCRIPTION

RECORD Number & TRACE Number

"*EDMT"-Procedure allows user to apply surgical mute to the data set based on Trace Header Indices for RECORD and TRACE Numbers. For example,

```
*EDMT  92  8   --> Use Source Sequence Number as RECORD Number
                   and Recording Channel number as TRACE Number;
*EDMT  12 14   --> Use CDP Number as RECORD Number
                   and CDP-Trace Number as TRACE Number.
```

Both Trace Header Indices are used for RECORD & TRACE Numbers in MUTE-Card.

Interpolation

Linear interpolation is used both within the RECORD and across the data set.
Therefore,

```
*EDIT  92  8
MUTE  100   1 20 30
```

will mute channel #1 and all other channels from 20 ms to 30 ms for record #100 and all other records likewise.

WITHIN EACH RECORD, interpolation is based on the slope between the specified TRACES, whereas beyond the specified trace the slope becomes zero. For example,

```
*EDIT 92 8
MUTE 100    5 40 80   15 60 90
```

will mute Channel #1 to #5 from 40 ms to 80 ms,
Channel #15 to #24 from 60 ms to 90 ms,
and apply linear interpolated mute between Channel #6 and #14.
Note also for this example that the mute-schedule is apply to Record #100 and all others.

The same principle is also applied ACROSS THE RECORDS. For example,

```
*EDIT  92 8
MUTE 45    1 40 80    24 60 100
MUTE 75    1 30 70    24 50  90
```

will mute Records #45 and lower with the first mute-schedule,
mute Records #75 and higher with the second mute-schedule,
and apply interpolated mute channel by channel between Record #46 and #74.

TO ELIMINATE GLOBAL EFFECT OF INTERPOLATION, USER MUST ASSIGN MUTE STARTING TIME AND MUTE ENDING TIME TO HAVE SAME VALUE.

*FILT (digital FILTering)

Procedure Definition Card: "*FILT"

"*FILT" PROCEDURE DEFINITION CARD

Field	Code / Parameter	Default
0	***FILT** - Procedure Definition Code	NA
1	low cut frequency	NA
2	high cut frequency	NA
[3]	Filter type: 1 = Bandpass 2 = Bandreject	1
[4]	Notch filter option 0 = Notch out 1 = Notch in	1
[5]	Filter length (in number of points)	31
None		

"*FILT" PROCEDURE DESCRIPTION

Card-control program employs the algorithm for digital filtering of seismic data from interactive filtering program. (refer to Section 5 on "Enhanced Seismic Data Processing").

*NMOT (Normal MOveouT correction)

Procedure Definition Card: "*NMOT"
Procedure Data Card : "VELF" <Required, at least 1 card>

"*NMOT" PROCEDURE DEFINITION CARD

Field	Code / Parameter	Default
0	***NMOT** - Procedure Definition Code	NA
[1]	Allowable Sample Stretch Ratio (ASSR) (FLOATING POINT) NOTE: Greater value gives deeper mute. Maximum ASSR = 0.99(allow up to 1% stretch) Minimum ASSR = 0.2(allow up to 400% stretch) Default = 0.5(allow up to 100% stretch)	0.5
None		

"VELF" PROCEDURE DATA CARD (Card[s] to follow *NMOT-Card)

Field	Code / Parameter	Default
0	**VELF** - VELocity-Function Procedure Data Code	NA
1	CDP number of this function	NA
2	Time (ms) of 1st time-velocity pair	NA
3	Velocity (length/s) at time in FIELD 2	NA
[4]	Time (ms) of 2nd time-velocity pair	
[5]	Velocity (length/s) at time in FIELD 4	
.	.	
.	.	

"*NMOT" & "VELF" ERROR MESSAGES

1. VELF: CDP# (Field# 1) must increase
2. No velocity function for *Nmot
3. No time-velocity pair
4. Negative time
5. Time > Processing trace length
6. Time must increase
7. Negative velocity
8. Allow only one *NMOT per pass
9. No field-geometry (*SORT) has been assigned

"*NMOT" PROCEDURE DESCRIPTION

Normal Moveout Correction & Allowable Sample Stretch Ratio

A NMO correction is applied to the seismic data to compensate for the dynamic time variations that result from changes in distance between the source and each receiver. The procedure uses the exact normal moveout equation

$$\text{Delta T} = \left(\frac{X^2}{V^2} + T_0^2 \right)^{1/2} - T_0 ,$$

where

Delta T = NMO correction,
X = shot-to-receiver distance,
V = stacking velocity, and
T_0 = zero-offset reflection time.

As a result, the NMO corrected data will appear as if the source-receiver offset is zero (vertical incidence).

The procedure employs linear interpolation (along time axis) for the calculated NMO correction generally falling between two successive samples.

The NMO process time stretches the data. It is therefore necessary to eliminate the distortion caused by NMO stretching. This is done by defining an allowable sample stretch ratio (ASSR).

By definition,
ASSR = ratio of separation in time between two successive samples before and after NMO correction.

Mathematically, ASSR can be expressed as

$$\text{ASSR} = T_0 \Big/ \left(\frac{X^2}{V^2} + T_0^2 \right)^{1/2} ,$$

which is an approximated function of shot-to-receiver distance, stacking velocity, and zero-offset time. Its theoretical value is between 0 and 1.

Examples for usage of ASSR as a limit for NMO mute:

ASSR = 0.2 implies > 400% stretching from original data is muted.
0.5 " > 100% " "
0.8 " > 25% " "

The NMO-procedure program applies the NMO mute only for the first (top) velocity layer. This is practically sufficient. since, mostly, only the near-surface data is critically stretched. Also, most velocity functions have increasing velocity with depth.

Maximum and minimum values of ASSR are set to be 0.99 and 0.2, respectively. User's supply value out of this range will be reset to the nearest border limit. If not specified, the default value is 0.5.

"VELF" PROCEDURE DATA DESCRIPTION

Velocity Function

1. CDP numbers used in specifying velocity functions must be in increasing order.

2. Specified times of time-velocity pairs within each velocity function must be increasing.

3. Linear interpolation between successive velocity functions is used in both the vertical (time) and horizontal (distance) domains.

For vertical interpolation, velocity is interpolated only within a specified time-velocity pair. Outside the range of a time-velocity pair, velocity is treated as constant.

For example, assume assignment of velocities 200, 300, 400 ft/s at time 20, 50, 80 ms at CDP #150.

```
*NMOT
VELF  150    20 200   50 300   80 400
```

Velocity interpolation (see also diagram chart below) will be linearly
between 200 ft/s at time 20 ms and 300 ft/s at time 50 ms;
and between 300 ft/s at time 50 ms and 400 ft/s at time 80 ms.
Velocity is a constant of 200 ft/s from time 0 to 20 ms; and 400 ft/s from time 80 ms downward.

```
 Time       Velocity
  0 ms-----------------------------------
          Constant 200 ft/s
 20 ms------200 ft/s---------------------
          Interpolation
          between 200 ft/s and 300 ft/s
 50 ms------300 ft/s---------------------
          Interpolation
          between 300 ft/s and 400 ft/s
 80 ms------400 ft/s---------------------
          Constant 400 ft/s
                 .
                 .
```

The same principle is also applied to horizontal interpolation.

*RESE (RESidual static corrEction)

Procedure Definition Card: "*RESE"
Procedure Data Card: "PERC" [Optional, 1 card]
"GATE" <Required, at least 1 card>

"*RESE" PROCEDURE DEFINITION CARD

Field	Code / Parameter	Default
0	***RESE** - Procedure Definition Code	NA
1	Static Limit in ms	NA
[2]	Number of summing CDPs to form pilot trace for cross-correlation	3
[3]	Number of Gates to apply	1
[4]	Cross-correlation coefficient threshold (FLOATING POINT)	0.5
[5]	MINIMUM offset to accept in forming pilot traces (FLOATING POINT)	0
[6]	MAXIMUM offset to accept in forming pilot traces (FLOATING POINT)	32000
None		

"PERC" PROCEDURE DATA CARD (OPTIONAL card to follow *RESE-Card)

Field	Code / Parameters	Default
0	**PERC** - PERCentage Procedure data code	NA
1	Percentage of first CDP forming pilot trace	NA
2	Percentage of Second CDP forming pilot trace	NA
.	.	.
.	.	.

NOTE: PERC-Card indicates weights in summing CDPs to form pilot trace. If this card is absent, will apply equal weight to all CDPs.
If present, user MUST specified number of percentage-values as FIELD 2 of *RESE-Card.
For example, let FIELD 2 of *RESE card = 7 (use 7 CDPs to form pilot trace), then

PERC 30 50 70 100 80 60 40

will apply weights of
100% to the CDP of interest,
30%, 50%, and 70% to the lower three CDPs, respectively,
80%, 60%, and 40% to the higher three CDPs, respectively.

"GATE" PROCEDURE DATA CARDs (Card[s] to follow *RESE-Card)

Field	Code / Parameter	Default
0	**GATE** - GATE-center-time procedure data code	NA
1	CDP number	NA
2	Gate-center-time in ms of the FIRST Gate	NA
3	Gate Size in ms of the FIRST Gate	NA
4	Gate-center-time in ms of the SECOND Gate	
5	Gate Size in ms of the SECOND Gate	
.	.	
.	.	

NOTE:
1. Interpolation of gate center time BETWEEN specified CDPs will be applied for more than one GATE-Cards.
2. Number of GateTime-GateSize pairs in GATE-Card MUST be as assigned in FIELD 3 of *RESE-Card.

"* RESE" & "GATE" ERROR MESSAGES

1. Minimum Offset >= Maximum Offset

2. Missing GATE-card for *RESE operation

3. Wrong # of Gate time-size pairs (predefined in *RESE-card =..)
   --- i.e. # of Gates (per GATE-card) DOES NOT match ..

4. GATE: CDP must be increasing

5. GATE: GateCenter - HalfGateSize - StaticLimit < 0");

6. GATE: GateCenter + HalfGateSize + StaticLimit > .. ms

7. *RESE: No time-gate pair

*RESE WARNING MESSAGES

1. Field# 2: Change CDPpilot to ..
   --- Change even number into odd number

2. Field# 4: Change threshold to 0.5
   --- Change threshold to default when it is out of range (0-1).

"*RESE" PROCEDURE DESCRIPTION

"*RESE"-Procedure straightforwardly applies residual static correction to each trace based on cross-correlation with pilot trace. It forces static shift to traces based solely on pilot trace. Therefore special care must be exercised when using *RESE-procedure.

Pilot Trace & Cross-Correlation

Pilot trace is formed by summing the surrounding CDPs. To form pilot trace, user can select Number of CDPs (FIELD 2, *RESE-Card) forming, acceptable Minimum Offset, and acceptable Maximum Offset (FIELD 5 and 6, *RESE-Card, respectively).

Cross-correlation coefficient (FIELD 4, *RESE-Card) can be between 0 (poorly match) and +1 (perfectly match). The default value is 0.5. The program will also report number of bad cross-correlation of which the resulting coefficient is lower than this threshold value.

Determination of Static Correction

Static correction is determined by cross-correlation of each input trace with the pilot trace within specified window (gate) time. In case of more than one windows or gates are specified, statics from all windows will be averaged.

Algorithm

If the input data is CDP NMO-corrected, the procedure first stacks the data (stored in disk file called "XXSTACK.TMP", under the same directory of the program run). If the input data is CDP stacked, this first step is skipped.

Based on the CDP-stacked file, it then forms the pilot trace file called "XXPILOT.TMP".

Next, each pilot trace is used as a reference to determine static shift by cross-correlation with input trace. The determined static is then applied to correct that input trace.

*RSRT (ReSoRTing field-geometry assigned seismic data)

Procedure Definition Card: "*RSRT"

"*RSRT" PROCEDURE DEFINITION CARD

Field	Code / Parameter	Default
0	***RSRT** - Procedure Definition Code	NA
[1]	Trace Header Index for RECORD Number (NOT limited as FIELD 1 of "*SORT" card)	12
[2]	Trace Header Index for TRACE Number within each record. (NOT limited as FIELD 2 of "*SORT" card)	0
[3]	Output trace DIRECTION flag +1 = ASCENDING; -1 = DESCENDING	1
None		

Updated header words: 1, 3, 4, 5, 14, 88.

"*RSRT" PROCEDURE DESCRIPTION

"*RSRT" procedure is designed to resort seismic data that has already been assigned field-geometry with "*SORT" procedure. In other words, starting with raw field data, "*SORT" procedure must be used to assign field-geometry and sort seismic data into one of 6 types (see "*SORT" procedure description). After that, user can apply "*RSRT" procedure in order to resort data into any form.

Please note that both RECORD Number Index and TRACE Number Index (FIELDs 1 and 2 of "*RSRT" Card) specified in "*RSRT" procedure are not limited as in "*SORT" procedure.

Please note also that user cannot use "*RSRT" to sort raw-field data, since raw field data has not been assigned field-geometry (instead, "*SORT" must be used there).

*SCAL (automatic gain control SCALing)

Procedure Definition Card: "*SCAL"

"*SCAL" PROCEDURE DEFINITION CARD

Field	Code / Parameter	Default
0	***SCAL** - Procedure Definition Code	NA
[1]	Window length (ms) *Default = 1/4 of input record length	*
[2]	Scaling option: 1 = Absolute Value Mean 2 = Root Mean Square	1
[3]	Reference Mean (RM) for data amplitude	50
None		

"*SCAL" ERROR MESSAGES

1. SCAL AGC--- window length too small!!
   --- Number of samples in the window is less than 5

2. SCAL AGC--- window length too large!!
   --- Number of samples in the window is larger than input record length to be processed.

3. *SCAL: option = 1 or 2 only

"*SCAL" PROCEDURE DESCRIPTION

Scaling is used to normalize to amplitude of a seismic trace. AGC scaling simulates feedback normalization by applying a variable gain factor to each individual sample based on the average amplitude of the surrounding data within the assigned window.

Scaling operation offers 2 options.

1) Absolute Value Mean.
The procedure uses the absolute-value-mean AGC applying at the center of the window with gain factor

$$G[n] = RM * W / \left(\sum_{i = n - w/2}^{i = n + w/2} | A[i] | \right)$$

2) Root Mean Square.
The procedure uses the root-mean-square AGC applying at the center of the window with gain factor

$$G[n] = RM / \left(\left(\sum_{i = n - w/2}^{i = n + w/2} (A[i])^2 \right) / W \right)^{1/2}$$

where, for both options,
G[n] = gain factor of the n(th) sample (which is at the center of window),
RM = Reference Mean (set to be 50 for 16-bit data),
W = fixed window length in sample unit,
and A[i] = original seismic data amplitude of the i(th) sample.

*SORT (SORTing seismic data)

Explanation: *SORT-Procedure require raw-field data as input

Procedure Definition Card: "*SORT"
Procedure Data Card: "PTRN", "PN"
"SHOT", "SN" and/or "SNSN"
"TABL" [Optional]

Explanation of Data Cards:
1. PTRN-procedure data cards: to define shot-receiver pattern.
2. SHOT-procedure data cards: to define shot-station location.
3. TABL-procedure data card: to list the resulting sorted table.

Rule of Order of Data Cards:
1. Required cards are PTRN-procedure data cards and SHOT-procedure data cards.
   PTRN-procedure data cards must precede SHOT-procedure data cards.
2. Optional card is TABL-procedure data card.
   If present, TABL-procedure data card must follow SHOT-procedure data cards.

"*SORT" PROCEDURE DEFINITION CARD

Field	Code / Parameter	Default
0	***SORT** - Procedure Definition Code	NA
[1]	Trace Header Index for RECORD Number Allowed values: 8, 12, 19, 86, 87, 92 (Default = 12)	12
[2]	Trace Header Index for TRACE Number within each record. Allowed values are: 0, 8, 12, 19, 86, 87, 92 (Default = 0 = input order)	0
[3]	Output trace DIRECTION flag +1 = ASCENDING; -1 = DESCENDING (Default = +1)	1
None		

Updated header words: 1, 3, 4, 5, 10, 12, 14, 19, 35, 82, 83, 84, 85, 86, 87, 88.

"*SORT" PROCEDURE DESCRIPTION

1. Sorting procedure can sort data into 6 types based on specified RECORD number in FIELD 1 of *SORT Card:
   - 8 = Common recording channel,
   - 12 = Common Depth Point (CDP),
   - 19 = Common offset,
   - 86 = Common receiver station number,
   - 87 = Common shot station number,
   - 92 = Common source sequence number

   RECORD numbers in the sorted output data will always be from the smallest RECORD number to the largest RECORD number, regardless of shot direction of input field data.

2. For TRACE Number within each RECORD, allowed Trace Header Index (FIELD 2 of *SORT Card) are 0, 8, 12, 19, 86, 87, and 92.
   Direction of TRACE Number is indicated with Trace DIRECTION Flag (FIELD 3 of *SORT Card). It can be either +1 (ascending order) or -1 (descending order).

3. For default in sorting (RECORD Number Index = 12, TRACE Number Index = 0, and Trace DIRECTION Flag = +1):
   Traces within each CDP gather will be in "input order." This means that whatever trace comes first in the input field data appears first in the CDP gather.

4. For multiple shots at the same shot station, the program is capable of separating each shot into different output files within a single pass. This can done be proper assignment of parameter in FIELD 6 of SN-CARD or SNSN-CARD.

"PTRN" & "PN" PROCEDURE DATA CARDS

"PTRN" CARD (Card to follow *SORT-Card)

Field	Code / Parameters	Default
0	**PTRN** - PaTteRN Procedure Data Code	NA
1	Station Spacing (can be FLOATING POINT) in length unit (eg. meters, feet, etc.) between successive stations	NA
[2]	Total recording Channels for this data set	24
[3]	Length Unit : 0 = FEET 1 = METERS else = undefined	0
None		

"PN" CARD (Card[s] to follow PTRN-Card) - Pattern Number

Field	Code /Parameter	Default
0	**PN** - Pattern-Number Procedure Data Code	NA
1	Pattern Number to be defined for this PN-card (Use 1 if only one pattern is defined, i.e. only one PN-card for this sort-deck.)	NA
2	Shot-station Number for this pattern	NA
3	Receiver-station Number of FIRST Channel for FIRST group of consecutive channels	NA
4	Number of consecutive channels to define for this FIRST group of channels	NA
5	Station increment between consecutive channel	NA
6	Receiver-station Number of FIRST Channel for SECOND group of consecutive channels	
7	Number of consecutive channels to define for this SECOND group of channels	
8	Station increment between consecutive channel	
.	.	
.	.	
.	.	

NOTE:

PN-CARD:

Example below defines 2 shot-receiver patterns for 24 channels data set with .25 meters of station spacing. Both patterns have shot locate at Station# 100.

```
First   PN  card  is  for end-on pattern  with
    Channel  1  at Station# 101
       "     2      "     102
       "     .      "      .
       "     .      "      .
       "    24      "     124
Second  PN card is for split-spread pattern with
    Channel  1  at Station# 120
       "     2      "     119
       "     .      "      .
       "     .      "      .
       "    12      "     109
       "    13      "      91
       "    14      "      90
       "     .      "      .
       "     .      "      .
       "    24      "      79
```

```
PTRN    .25
PN     1   100      101  24  1
PN     2   100      120  12 -1    91 12 -1
```

"PTRN" & "PN" ERROR MESSAGES

1. Total_channels in PTRN-card (..) not the same as input data (..)
2. Missing Ptrn-card
3. PTRN: Require Field# 1
4. PTRN: Invalid recording channels at Field# 2
5. Missing PN-card
6. PN: Repeated pattern number (Field# 1)
7. PN: Defined channels exceed ..
8. PN: .. channels have been defined, but .. channels are required
9. PTRN: Cannot find 16-bit multiplier for Field# 1
   --- problem relates to floating point number
10. Pattern # .. has not been defined in PN-card

"SHOT" & "SN" and/or "SNSN" PROCEDURE DATA CARDS

Explanation:
1. Cards to follow SHOT-CARD can be SN-CARD[s] only, or SNSN-CARD[s] only, or a mixture of them.
2. ONE SN-CARD will be used to define shot location for ONE field record (or ONE source sequence number).
3. ONE SNSN-CARD will be used to define shot locations for a set of MORE-THAN-ONE field records.

"SHOT" CARD (Prerequisite: PTRN-Card & PN-Card[s])

Field	Code / Parameter	Default
0	**SHOT** - Procedure Data Code	NA
None		

"SN" CARD (Card[s] to follow SHOT card)

Field	Code / Parameter	Default
0	**SN** - Procedure Data Code	NA
1	Source Sequence Number (SSN)	NA
2	Shot-station Number for this SSN (in FIELD 1)	NA
[3]	Pattern Number associated with this SSN	1
[4]	In-line offset (can be FLOATING POINT) (POSITIVE VALUE if shot between station in FIELD 1 and next higher station-number; NEGATIVE value if shot between station in FIELD 1 and next lower station-number)	0
[5]	Off-line (perpendicular) offset (can be FLOATING POINT)	0
[6]	Repeated-shot number (at the same shot-station) associate with this SSN	1
[7]	FIRST Channel to OMIT for this field record	*
[8]	LAST Channel to OMIT for this field record (Leave BLANK for non-omitting; * Default = non-omitting)	*
. . .	. . .	

"SNSN" CARD (Card[s] to follow SHOT-card)

Field	Code / Parameter	Default
0	**SNSN** - Procedure Data Code	NA
1a	"<" - Open-Angle-Bracket Symbol	NA
1b	Starting Source Sequence Number (SSN)	NA
1c	Ending SSN	NA
1d	Incrementing value of SSN (must be POSITIVE value only, since the expects SSNs to be in ascending order)	NA
1e	">" - Closed-Angle-Bracket Symbol	NA
2a	"<" - Open-Angle-Bracket Symbol	NA
2b	Starting Shot-Station Number associated with specified SSN in FIELD 1b	NA
2c	Ending Shot-station Number associated with specified SSN in FIELD 1c	NA
2d	Incrementing (POSITIVE value) or DECREMENTING (NEGATIVE value) value of Shot-station number	NA
2e	">" - Closed-Angle-Bracket Symbol	
[3]	Pattern Number associated with this group of SSNs specified in FIELD 1	1
[4]	In-line offset (can be FLOATING POINT) (POSITIVE VALUE if shot between station in FIELD 1 and next higher station-number; NEGATIVE value if shot between station in FIELD 1 and next lower station-number)	0
[5]	Off-line (perpendicular) offset (can be FLOATING POINT)	0
[6]	Repeated-shot number (at the same shot-station) associate with this group of SSNs specified in FIELD 1	1
[7]	FIRST Channel to OMIT for this field record	*
[8]	LAST Channel to OMIT for this field record (Leave BLANK for non-omitting; * Default = non-omitting)	*
.	.	
.	.	
.	.	

NOTE:

SN-CARD:

1. Source Sequence Numbers (FIELD 1 of SN-card) for successive SN-CARDs must be in ASCENDING order, but DOES NOT HAVE TO BE SUCCESSIVE. MISSING SSN[s] WILL BE AUTOMATICALLY OMITTED, i.e. not include in the output sorted data.

2. Shotpoint station locations (FIELD 2 of SN card) identified on successive SN cards can be in ascending or descending order or any combination of both both, i.e. next-shotpoints can move from low-number to high-number stations, or from high-number to low-number stations, or a mixture of them within the data set to be sorted.

3. Pattern Number (FIELD 3 of SN-card) must be corresponding to the pattern number specified earlier in FIELD 1 of PN-card.

4. Repeated-shot number is for the case of more than one shot per station. Default = 1. Use the same number through out SN-carddeck if there is only one shot per shot-station.

5. To omit more than one group of consecutive recording-channels, use FIELDs 9 & 10, 11 & 12, ...

SNSN-CARD:

SNSN-CARD is identical to SN-CARD except FIELDs 1 & 2, which are used to define a group of Source Sequence Numbers.

Examples

The following 2 sets of SN-Cards and SNSN-Card yield the same result.

```
First set:      SHOT
                SN   11   101
                SN   12   100
                SN   13    99
                SN   14    98
                SN   15    97

Second set:     SHOT
                SNSN  < 11  5  1 >   < 101  97  -1 >
```

"SHOT" & "SN" & "SNSN" ERROR MESSAGES

1. SNSN: Field# 1a must be "<"
2. SNSN: Field# 1e must be ">"
3. SNSN: Field# 2a must be "<"
4. SNSN: Field# 2e must be ">"
5. SNSN: Wrong SSN at Field# 1b or 1c
6. SNSN: Field# 1d must be positive
7. SNSN: Field# 1b, 1c, and 1d do not match
8. SNSN: Field# 1 (SSNs) and 2 (Shots) do not match
9. Missing SHOT-card
10. SHOT: No parameters required
11. SN: SSN (Field# 1) must increase
12. SNSN: SSN (Field# 1b) must increase
13. Missing SN- or SNSN-card
14. Omitted channel (Field# ..) > total_channel (..)
15. First (Field# ..) > Last (Field# ..) omitted channel
16. Total omitted channels > ..
17. Cannot find 16-bit multiplier for in-line offset
18. Cannot find 16-bit multiplier for perpendicular offset

"TABL" PROCEDURE DATA CARD (to list sorted TABLe) [OPTIONAL]

(Optional Card to follow suite of SHOT-procedure data cards)

Field	Code / Parameter	Default
0	**TABL** - sorted-TABLe Procedure Data Code	NA
[1]	Flag to print Receiver Diagram 1 = print; 0 = no print (Default = 1)	1
[2]	Flag to print Sorted Table 1 = print; 0 = no print (Default = 1)	1
None		

NOTE: Upon defining this card, the program will list the resulting sorted-table. This card is useful especially when user wants to check for correctness of the input sort-deck without actual sort of seismic data. To do this, the *INPF card and *OUTF card can be omitted.

The resulted sorted table will be printed to deck-output (disk) file.
For example, after created sort deck in file "INLIST.LST", user can run the test by typing

```
SEIS INLIST.LST OUTLIST.LST
```

with carriage return.
After the run is complete, user can type

```
TYPE OUTLIST.LST
```

and hit carriage return to view the sort-table on the terminal screen (use CTRL-S to freeze the screen, and another CTRL-S to let the screen roll);
or user can type

```
COPY OUTLIST.LST PRN
```

and hit carriage return in order to obtain a hard copy from the dot-matrix printer. (Make sure that the printer -power is on.)

*STAK (trace STAcKing)

Procedure Definition Card: "*STAK"

"*STAK" PROCEDURE DEFINITION CARD

Field	Code /Parameter	Default
0	***STAK** - Procedure Definition Code	NA
[1]	Stack option 1 = actual folds 2 = use square root of actual folds	1
None		

NOTE: Updated header word : 1, 17, 88

"*STAK" PROCEDURE DESCRIPTION

The CDP-stacking procedure employs an averaging scheme that sums all seismic traces of a particular CDP and then divides by the number of traces summed (Stack option = 1) or square root of the number of traces summed (Stack option = 2).

With proper normal moveout (NMO) corrections previously applied, CDP stacking functions as a filter to enhance the signal-to-noise ratio.

*STAT (apply STATic correction from header word)

Procedure Definition Code: "*STAT"

"*STAT" PROCEDURE DEFINITION CARD

Field	Code /Parameter	Default
0	***STAT** - Procedure Definition Code	NA
1	Option 1 = Datum statics correction from trace header 2 = Automatic statics correction from AUTS-table 3 = bulk statics	NA
2	If Field #1 is 1, leave blank. If Field #1 is 2, enter AUTS-table filename (table of surface-consistent residual source & receiver static correction). If Field #1 is 3, enter bulk statics in ms.	NA
None		

NOTE: Updated header word : 52.
(Also update word 89 and 90 if FIELD #1 is 2)

*STAT ERROR MESSAGE

1. *STAT: Missing AUTS-table filename in Field# 2
2. *STAT: Missing bulk static shift (ms) in Field# 2
3. *STAT: Field# 1 is invalid
4. Format error in input AUTS-table file ...
   This is an unusual error -- CHECK DISK SPACE,
   IF PROBLEM CAUSE IS NOT FOUND, REPORT PROBLEM TO PROGRAMMER!
5. AUTS-table file ... does not exist

"*STAT" PROCEDURE DESCRIPTION

Datum Static Corrections

If FIELD #1 is 1, the procedure applies static shift to the data based on information in trace-header word 50 -- for source static, and word 51 -- for receiver static. Total static shift applied to each trace will be stored in word 52.

The amount of static correction must have been previously calculated. For example, datum static correction can be calculated earlier during CDP sorting, while the actual data shift has not been done. The amount of source and receiver static corrections would be stored in trace-header word 50 and 51, respectively, for each seismic trace. By applying the *STAT procedure, the actual (physical) data shift is accomplished. Total shift (sum of source and receiver static shifts) applying to each trace will then be recorded in trace-header word 52.

Residual Static Corrections (from input static-table file)

If FIELD #1 is 2, user has to enter static-table filename in FIELD #2.

Bulk Static Correction

If FIELD #1 is 3, user has to enter bulk static in ms in FIELD #3.

Sign convention for static shift

POSITIVE value implies shifting DOWN away from zero-time.
NEGATIVE value implies shifting UP toward zero-time.

*SURF (calculation of SURFace elevation correction)

Explanation: *SURF-Procedure must be applied either
A) after *SORT-Procedure, within the same run, or
B) to the data set that has been sorted.

Procedure Definition Card: "*SURF"
Procedure Data Card: "ALVF" <Required, 1 card>
"SE" <Required, at least 1 card>

"*SURF" PROCEDURE DEFINITION CARD

Field	Code / Parameters	Default
0	***SURF** - Procedure Data Code	NA
1	Datum elevation	NA
None		

"ALVF" CARD (Card to follow SURF card)

Field	Code / Parameter	Default
0	**ALVF** - Average Layered-Velocity Function Procedure Data Code	NA
1	Surface elevation	NA
2	Average velocity from surface elevation in FIELD 1 down to elevation in FIELD 3	NA
3	Elevation of FIRST subsurface layer	NA
[4]	Average velocity from surface elevation in FIELD 1 down to elevation in FIELD 5	
[5]	Elevation of SECOND subsurface layer	
.	.	
.	.	

"SE" CARD (Card[s] to follow ALVF-Card)

Field	Code / Parameter	Default
0	**SE** - Station-Elevation Procedure Data Code	NA
1	Station number to define shot & receiver elevations	NA
2	Elevation of source at this station	NA
[3]	Elevation of receiver at this this station * if not specified, use value in FIELD 2.	*
None		

NOTE:

ALVF-Card

1) Please note the average velocity in ALVF card that it is the average velocity (NOT interval velocity) from surface elevation down to the specified level.

2) *** WARNING:

Surface elevation (FIELD 1, ALVF-Card) must be higher than (or at least equal to) datum elevation and shot & receiver elevations in SE-Cards.

The last (lowest) specified elevation in ALVF-Card must be less than (or at most equal to) datum elevation and shot & receiver elevations in SE-cards.

SE-Card

The SE card will assume that both source and receiver elevations are the same if only FIELD 3 is specified.

"SURF" & "ALVF" & "SE" ERROR MESSAGES

1. Missing ALVF-card for *SURF-operation
2. Surface Elevation < Datum (..)
3. ALVF: Elevation must be decreasing
4. Calculated interval velocity is negative
5. Lowest elevation > Datum (..)
6. Missing SE-card for *SURF-operation
7. Station # must be successively increasing
8. Shot Elevation > Highest ALVF-Elevation (..)
9. Shot Elevation < Lowest ALVF-Elevation (..)
10. Receiver Elevation > Highest ALVF-Elevation (..)
11. Receiver Elevation < Lowest ALVF-Elevation (..)
12. Receiver Station # .. has not been define in SE-card
13. Shot Station # .. has not been define in SE-card

"*SURF" PROCEDURE DESCRIPTION

Usage of *SURF-Procedure

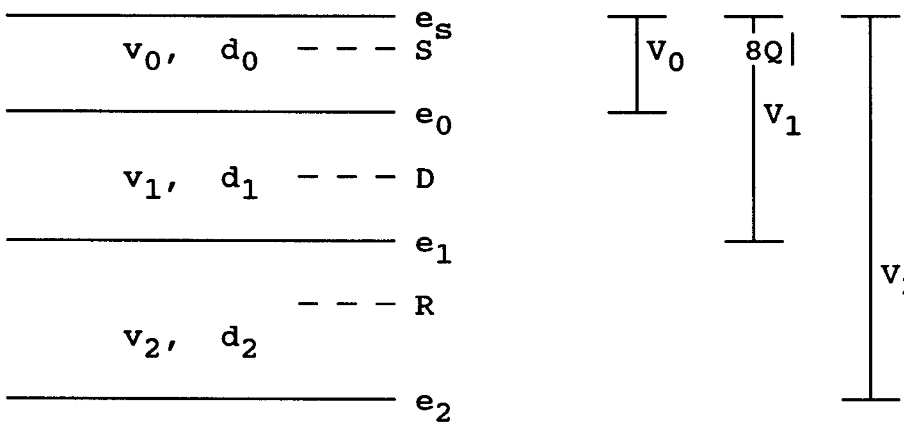
*SURF-Procedure ONLY calculates elevation correction (in ms) and stores in Trace Header Word (#50 for Source, #51 for Receiver). It does not actually apply static correction to input traces. User can apply *STAT-Procedure in order to accomplish the static correction either after *SURF-Procedure within the same run or later on in a different run.

*SURF-Procedure requires data that have been assigned shot & receiver station-numbers. Therefore, if input data is raw-field data, *SORT-Procedure must be applied preceding *SURF-Procedure within the same run. However, if input data has been assigned field-geometry in the previous run, *SURF-Procedure can be applied without requiring *SORT-Procedure.

Average Layered-Velocity Function

Refer to the diagram below.

Let D = datum elevation (FIELD 1, *SURF-Card)
e_s = surface elevation (FIELD 1, ALVF-Card)
V_0 = average velocity from e_s down to e_0 (FIELD 2, ALVF-Card)
e_0 = 0th-layer elevation (FIELD 3, ALVF-Card)
V_1 = average velocity from e_s down to e_1 (FIELD 4, ALVF-Card)
e_1 = 1st-layer elevation (FIELD 5, ALVF-Card)
V_2 = average velocity from e_s down to e_2 (FIELD 6, ALVF-Card)
e_2 = 2nd-layer elevation (FIELD 7, ALVF-Card)



*SURF-Procedure will first calculate the interval velocities based on the formula

$$v_i = \frac{V_i * d_i}{\sum_{n=0}^{i} (d_n) - V_i * \sum_{n=0}^{i-1} \left(\frac{d_n}{v_n}\right)}$$

where v_i = interval velocity of the ith-layer, (with $v_0 = V_0$),
and d_i = thickness of the ith-layer = $e_{i-1} - e_i$,
(with $d_0 = e_s - e_0$).

To calculate elevation static, suppose the source elevation is at S (FIELD 2, SE-Card) and receiver elevation is at R (FIELD 3, SE-Card). Then

$$\text{Source Static} = \frac{D - S}{\text{velocity}} = \frac{D - e_0}{v_1} + \frac{e_0 - S}{v_0}$$

$$\text{Receiver Static} = \frac{D - R}{\text{velocity}} = \frac{D - e_1}{v_1} + \frac{e_1 - R}{v_2}$$

5. ENHANCED SEISMIC DATA PROCESSING

The program, called "FMAIN" is a general purpose procedure to be used for enhancing seismic data. This program includes several functional modules - Filtering, Spectrum analysis, Deconvolution, Filter characteristic analysis, Data set comparison, and Manual static corrections. FMAIN also handles various types of seismic data - Raw-field data, SEG-Y field data, CDP sorted data, and CDP stacked data. The history of processing is written into the header of each seismic trace. All functions are performed in either an interactive mode or a batch mode. The FMAIN may be used before or after CDP processing.

IMPORTANT: In order to execute FMAIN, COMMAND.COM and FMAIN.HLP **must** exist in the working directory.

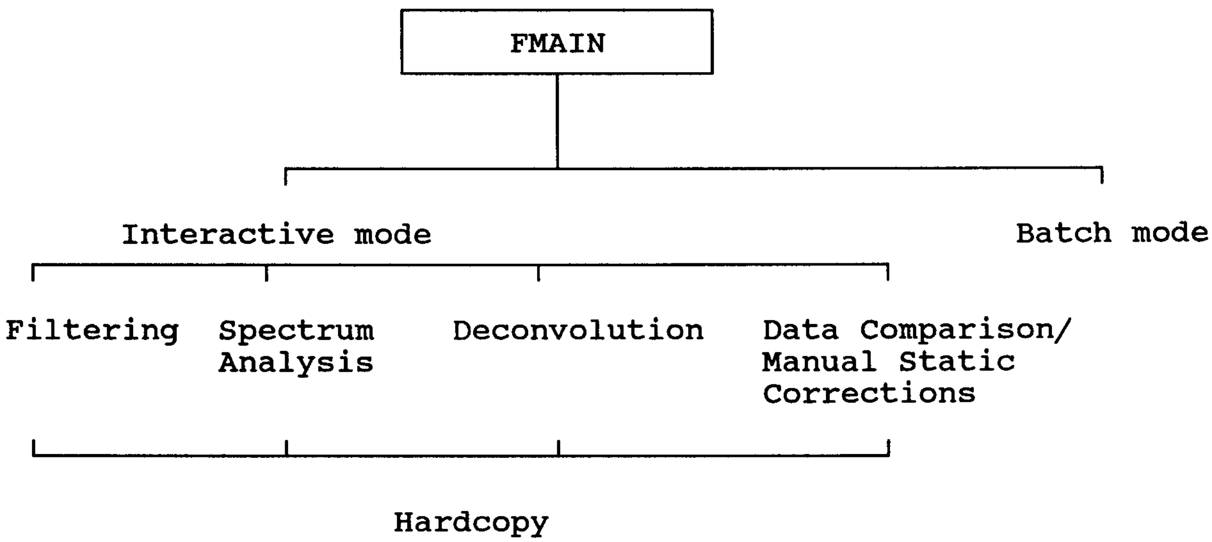


Figure 2. Organization of "FMAIN"

5.1 Filtering

A digital filter is established by entering a set of filter parameters - Low-cut and High-cut frequencies, and Notch in (or out), and then is applied to various types of seismic data. The filtered data is saved out to an output file. The filtering parameters are saved into the header of the seismic trace.

The filtering procedure is accomplished in two steps. (1) design of a filter and (2) filter application.

The designed filter is a zero-phase FIR(Finite Impulse Response) filter of 31 points. The basic approach to FIR filter design is to obtain a finite-length impulse response by truncating an infinite-duration impulse response sequence of the desired ideal filter with a proper taper. This design concept of the FIR filter is based upon "Digital Signal Processing" (Oppenheim and Schafer, 1975).

The actual filtering consists of many convolutions of seismic data with the designed filter coefficients. The convolution procedure uses the overlap-add method, wherein, the partial parts of a seismic trace are convolved with filter coefficients and then the overlapped sections are added. The convolution routine uses the Fast Fourier Transform.

A) Filter characteristics

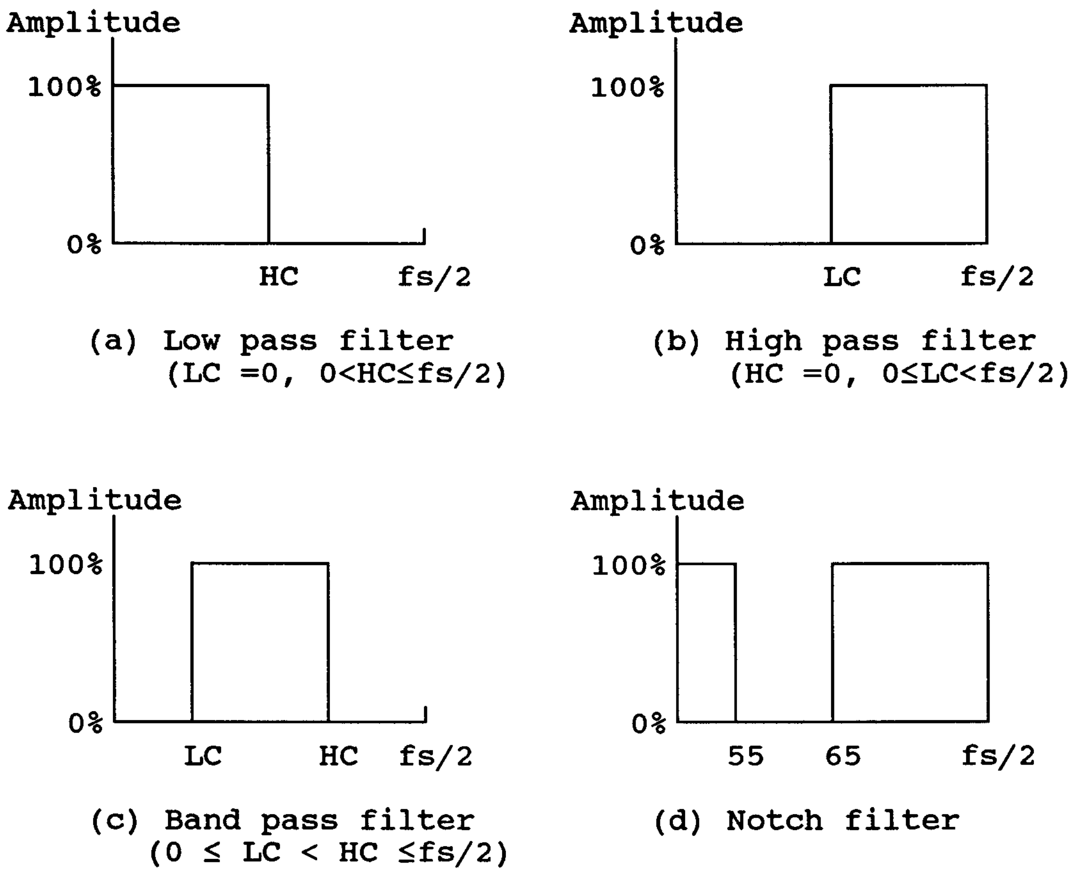
1) Filter types
   Low pass, High pass, Band pass, and Notch (60 & 120Hz) Filters

2) Filter length
   A variable-length zero phase FIR filter.
   (Typically 31 points)

3) Window
   Hamming window

B) Filter Definitions

Various types of IDEAL filters are defined by a set of input parameters - Low-Cut(LC), High-Cut(HC), and Notch frequencies.



Note: "fs" is a sampling frequency.

Figure 3. Various types of filters

In practice, of course, IDEAL filters are impossible to achieve. FMAIN allows the user to examine graphically the actual characteristics of a designed filter. See section 5.4.

C) Batch Filtering

Execution:

```
FMAIN F <input file> <output file> <p1><p2><p3><p4><p5>
```

where

Input file : Input file of Raw-field data, SEG-Y field data, CDP sorted data, or CDP stacked data.

Output file: Output file (The output file has the same data structure that the input file.)

p1 : Low cut frequency

p2 : High cut frequency

p3 : Notch In("1") or Out("0") [notch frequencies:55-65 Hz]

p4 : Starting source sequence number to be filtered or Starting CDP number to be filtered

p5 : Ending source sequence number to be filtered or Ending CDP number to be filtered

Example: Input file :"BRAD.DAT", Output file: "JUNK.DAT"
low-cut = 100(Hz), High-cut = 350(Hz), Notch out.
Starting SSN numbers to be filtered: 20 - 30

Command:

```
FMAIN F BRAD.DAT JUNK.DAT 100 350 0 20 30
```

D) Interactive Filtering

The interactive filtering shows a set of pictures on the monitor screen - pre-filtered data and post-filtered data Refer to the interactive commands for further details.

5.2 Spectrum Analysis

The spectrum analysis module provides the user with Fourier analysis capabilities. A seismic trace in the time domain is transformed into a series of spectra in the frequency domain while estimating the power spectra over a user defined time window. To avoid the Gibbs phenomena, a triangular time window is used.

Execution:

```
FMAIN S <input file> <output file> <p1><p2><p3><p4>
```

where

input file and output file: File name of Raw-field data, SEG-Y field data, CDP sorted data, or CDP stacked data.

(For most applications, the input file and output file are assigned to pre-filtered and post-filtered data files, respectively.)

p1 : Source sequence number (or CDP number) to be analyzed.

p2 : Trace number (must be 1 for CDP-stacked data file).

p3 : Starting time(msec) of window

p4 : Ending time (msec) of window

Example: Input file : "BRAD.DAT" (Raw-field data file)
Output file: "BRAD.FLT" (Filtered data file from BRAD.DAT)
SSN number : 21
Trace number : 7
Time window: 10 to 100 (msec)

Command:

```
FMAIN S brad.dat brad.flt 0 21 7 10 100
```

5.3 Deconvolution

The deconvolution routine removes multiples or reverberated signals from the data. This algorithm is programmed based upon the least-square predictive deconvolution method.

Execution:

FMAIN D <input> <output> <p1><p2><p3><p4><p5><p6><p7>

where

input file and output file: File name of Raw-field data, SEG-Y field data, CDP sorted data, or CDP stacked data.

p1 : Percentage of whitening (1 - 10%)

p2 : Prediction length(α)

If $\alpha = 0$, the prediction length is set to be the 2nd zero crossing point.
If $\alpha = 1$, spiking deconvolution will be performed.

Otherwise, α will be set manually.

p3 : Filter length

p4 : Start point for autocorrelation time window(msec).

p5 : End point for autocorrelation time window(msec).

p6 : Start SSN (or CDP number) to be processed.

p7 : End SSN (or CDP number) to be processed.

Example: Input file : "BRAD.DAT" (Raw-field data file)
Output file: "BRAD.DCN"
Whitening : 4 %
2nd zero-crossing point: $\alpha = 0$
Filter length : 21
Time window for autocorrelation : 5 to 45 (msec)
SSN : 21 - 22

Command:

```
FMAIN D brad.dat BRAD.DCN 4 0 21 5 45 21 22
```

5.4 Filter Characteristic Analysis

This routine shows filter characteristics designed by the operator's parameters. The designed filter may be applied to a real data set using filtering commands.

Execution:

```
FMAIN X <p1> <p2> <p3> <p4> <p5> <p6>
```

p1 : Sampling rate(Hz)

p2 : Low-cut frequency

p3 : High-cut frequency

p4 : Notch in["1"] or out["0"]

p5 : Filter length

p6 : command flag
If "1", make a hard copy.
If "2", pause for another control
Otherwise, exit

Example:

The following command show the analysis of a filter which has filter length 61, low-cut 100 Hz, high-cut 350, and notch out at the sampling rate of 1000Hz.

Command:

```
FMAIN X 1000 100 350 0 61
```

5.5 Data Set Comparison

The data set comparison allows the user to compare two sets of data files graphically. The two data files must have the same format. After invoked, all control commands to manipulate the data follow the interactive commands. Refer to section 5.7.

Execution:

FMAIN C <input file> <output file> <p1>

where

input/output files: Raw-field data, SEG-Y field data, CDP sorted data, or CDP stacked data.

<p1> : SS number or CDP number to be compared.

5.6 Manual Static Correction

The data manual static correction allows the user manually to change the time offset of a data set. After invoked, all control commands to manipulate the data follow the interactive commands. Refer to section 5.7.

Execution:

FMAIN M <input file> <output file>

where

input file: Input file name of Raw-field data, SEG-Y field data, CDP sorted data, or CDP stacked data.

output file: Working file name.
(The working file will be used for saving the results of manual static correction.)

5.7 Interactive Operations of FMAIN

FMAIN provides the user with a interactive mode that allows him to deal directly with the data sets. This is accomplished through the use of screen menus and graphical presentation of seismic data.

Execution: FMAIN <cr>

Interactive Command Summary

A. Main Menu

- function selection commands

F1: Filtering
F2: Spectra analysis
F3: Deconvolution
F4: Filter characteristic analysis
F5: Select data files
F6: Compare two sets of data
F7: Manual static correction
Alt-H: Help
Esc : Exit

A.1 Filtering

F1: Choose the graphic mode
F2: Compare the filtered/unfiltered data with graphics.
F3: Change filter parameters
F4: Run filtering
F5: Select data files
Esc : Exit to Main menu

A.2 Spectrum analysis

F3: Change operation parameters
F4: Run
F5: Select data files
Esc : Exit to Main menu

A.3 Deconvolution

F1: Choose the graphic mode
F2: Compare the filtered/unfiltered data with graphics.
F3: Change decon parameters
F4: Run
F5: Select data files
Esc : Exit to Main menu

A.4 Filter Characteristic Analysis

F3: Change filter parameters
F4: Run
Esc : Exit to Main menu

A.5 Manual static correction commands

F1: Mark start/end points
F2: Redraw
F3: Move a set of data using marked start/end points
Esc : Exit to Main menu

B. Graphic commands

arrow(->)	: Move to the next SS number(or CDP number)
arrow(<-)	: Move to the previous SS number(or CDP number)
arrow down	: Increase starting offset point by 50 samples
arrow up	: Decrease starting offset point by 50 samples
Ins	: Increase number of data bits shown
Del	: Decrease number of data bits shown
Home	: Normalize on/off
P	: Hard-copy
Esc	: Exit to the previous menu

APPENDIX A
BRIEF OVERVIEW OF MSDOS COMMANDS

1. COPY source.dat destiny.dat
   Copy content of source.dat to destiny.dat
   Old content of destiny.dat will be lost no matter what!!!!
   If destiny.dat did not exist, it will be created.
   Example:
   COPY SYNTH.DAT NEW.DAT - Copy content of SYNTH.DAT to NEW.DAT. Old content of NEW.DAT will be lost.
   COPY OUT.TXT PRN - Copy content of textfile to the printer.
   COPY CON PRN - Copy from console to printer.
   (Note: CON and PRN are preserved device names for console and printer, respectively.)

2. DIR - List current files in current directory.
   DIR *.dat - List current files having extension .dat
   Note that available space left will also be reported in bytes.

3. DEL junk.dat - Delete file junk.dat
   DEL *.TMP - Delete all file having extension .TMP

4. REN oldname.old newname.new - Rename filename

5. TYPE textfile.txt - Display content of text file on the screen.

6. G: - goto G: Drive (or change drive to G:)
   D: - goto D: Drive
   Note that drive name can be used to indicate specific drive for certain filename. For example,
   COPY D:TABLE.TBL G:TB.C
   will copy file TABLE.TBL from D: drive to file TB.C in G: drive.

7. Form of Filename is
   PRIMNAME.EXT
   The primary filename (PRIMNAME) consists of 1 to 8 characters and is REQUIRED. The extension (EXT) consists of 1 to 3 characters and is OPTIONAL. The primary name and extension must be separated by a PERIOD (.).

8. Control-Character Description

   CTRL-S : Freeze the screen (useful to view long textfile on the screen)
   To let the screen roll again, hit another CTRL-S.
   CTRL-C : Aborts current command or program run
   CTRL-BREAK : Aborts current command or program run immediately
   CTRL-ALT-DEL : Reboot the system (without power off the system)
   This should be used only in an unavoidable situation (such as when the system hangs up).

APPENDIX B

LIST OF RESERVED FILENAMES

The following filenames are reserved for processing.

FMAIN.HLP
OUTPUT.DAT
SAMPLE.DAT

BISCONV.EXE
FMAIN.EXE
PLOT.EXE
RESET.EXE
SEIS.EXE
TRHD.EXE
TRHDCHNG.EXE
VELP.EXE
VIEW.EXE
VSCN.EXE

COMMAND.COM

APPENDIX C
USER MANUAL GLOSSARY

Assr - Allowable sample stretch ratio
*Aued - Automated trace editing procedure
*Auts - Automatic static procedure
Alvf - Velocity function for surface elevation static
BISCONV - Bison field-data conversion program

*Edfm - Edit : first arrival mute
*Edkl - Edit : kill trace
*Edmt - Edit : surgical mute
>>End - End Marker of card deck

*Filt - Card-control filtering procedure
FMAIN - Interactive filtering, spectral analysis, and deconvolution package
Farm - First-arrival-muting procedure data card
Gate - Gate center time for residual static procedure
Gcen - Gate center time for automatic static procedure

*Inpf - Input data procedure
Kill - Kill trace procedure data card
Mute - Surgical-mute procedure data card
*Nmot - Normal moveout procedure
*Outf - Output data procedure

Perc - Percentage definition for pilot-CDP forming
PLOT - Plot, graphic display program
Ptrn - Field-geometry pattern procedure data card-control
Pn - Field-geometry pattern data card
*Rese - Residual-static-editing procedure
*Rsrt - Resort procedure

*Scal - AGC scaling procedure
Se - Surface elevation data card
SEIS - Card-control processing package
Shot - Shot procedure data card
Sn - Shot sequence number card
Snsn - Group of shot sequence number card
*Sort - Sort procedure
*Stak - CDP stack procedure
>>Start - Start Marker of card deck
*Stat - Static correction procedure
*Surf - Surface elevation procedure

Tabl - CDP listing procedure data card-control
Tapr - tapering procedure data card
TRHD - Program to dump chosen trace header words
Velf - Velocity function for NMO correction card-control
VELP - Velocity picking program
VSCN - Velocity scan analysis (CDP data) program

IN-FIELD SEISMIC DATA PROCESSING

By

C. Dao Somanas
Brett C. Bennett
Young J. Chung

Kansas Geological Survey
Lawrence, Kansas

SUMMARY

With the advances made in microcomputer technology, in-field seismic data processing now has become practical for engineering and ground-water applications. A hardware/software design orchestrates the down-loading of field data in 16-bit fixed-point format from seismic field tape to an IBM-PC/AT compatible microcomputer. The CDP processing software includes trace sorting, trace editing, static correction (including elevation, automatic, and residual), velocity analysis, normal moveout correction, CDP stacking, AGC scaling, filtering, spectrum analysis, deconvolution, and data display. Each processing operation has user-selectable parameters that can be defined according to specific needs and can operate in any sequence. A 100-shotpoint data set with 24 channels per shotpoint and 500 samples per channel can be processed to brute stack within 20 minutes.

SYSTEM COMPONENTS

IBM-PC/AT compatible microcomputer equipped with:

- 640 Kbytes of memory
- Microsoft DOS 3.1 or greater
- 1 or 2 1.2Mbytes/360Kbytes floppy disk drives
- 10 Mbytes or more of fixed disk storage
- 80287 math co-processor
- EGA color video interface card
- EGA color monitor

Epson compatible graphic printer

PROCESSES SUPPORTED

Field data transfer

Data display

Velocity analysis

- Velocity calculation from reflection event
- Velocity scan analysis of CDP data

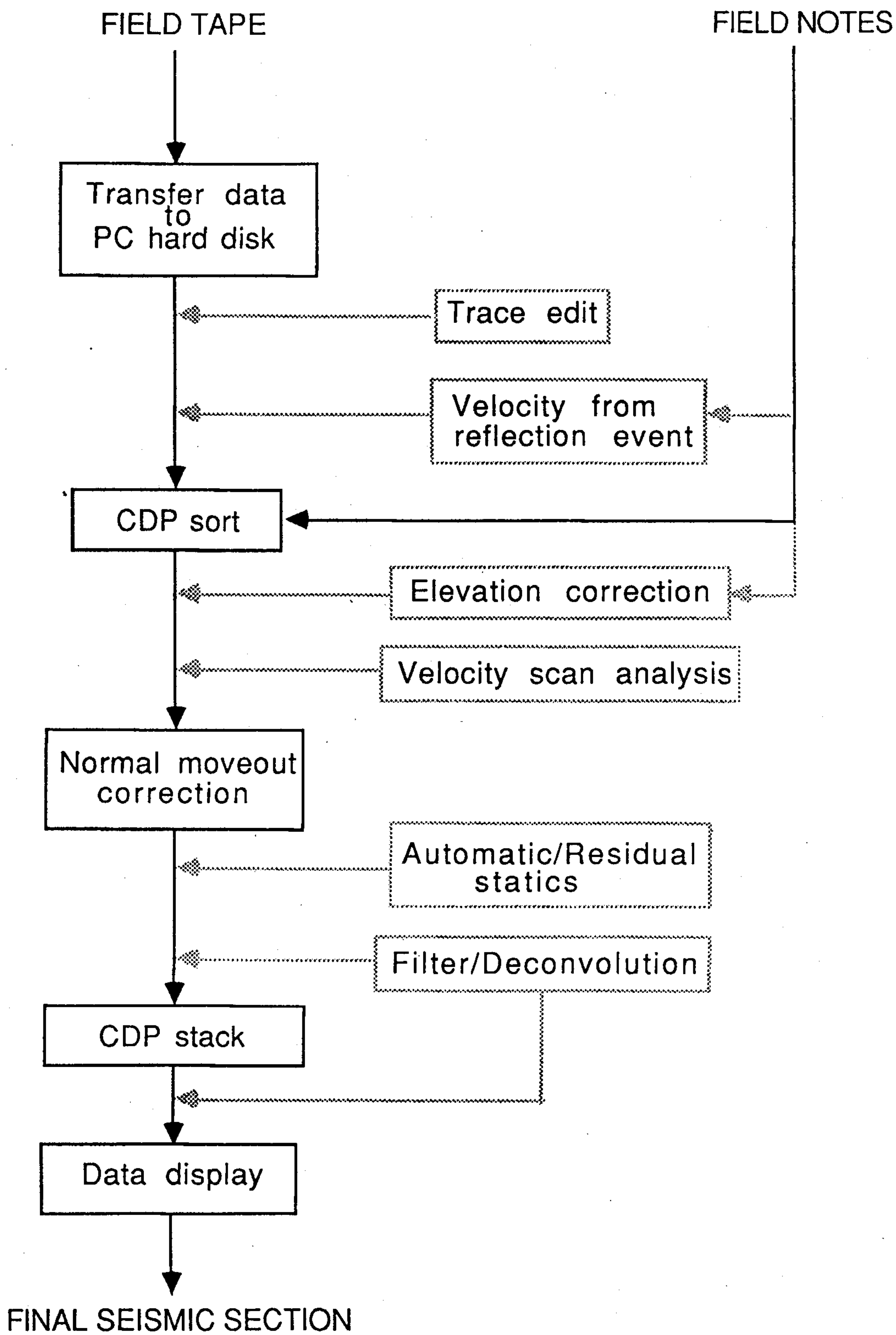
Card-control processing

- Trace sorting
- Trace editing
- Static correction (surface elevation, automatic, and residual)
- Normal moveout correction
- CDP stacking
- AGC scaling

Enhanced data processing

- Filtering
- Spectral analysis
- Deconvolution (spiking and 2nd-zero-crossing predictive)

BLOCK DIAGRAM OF PROCEDURE PROCESSING



FIELD DATA TRANSFER

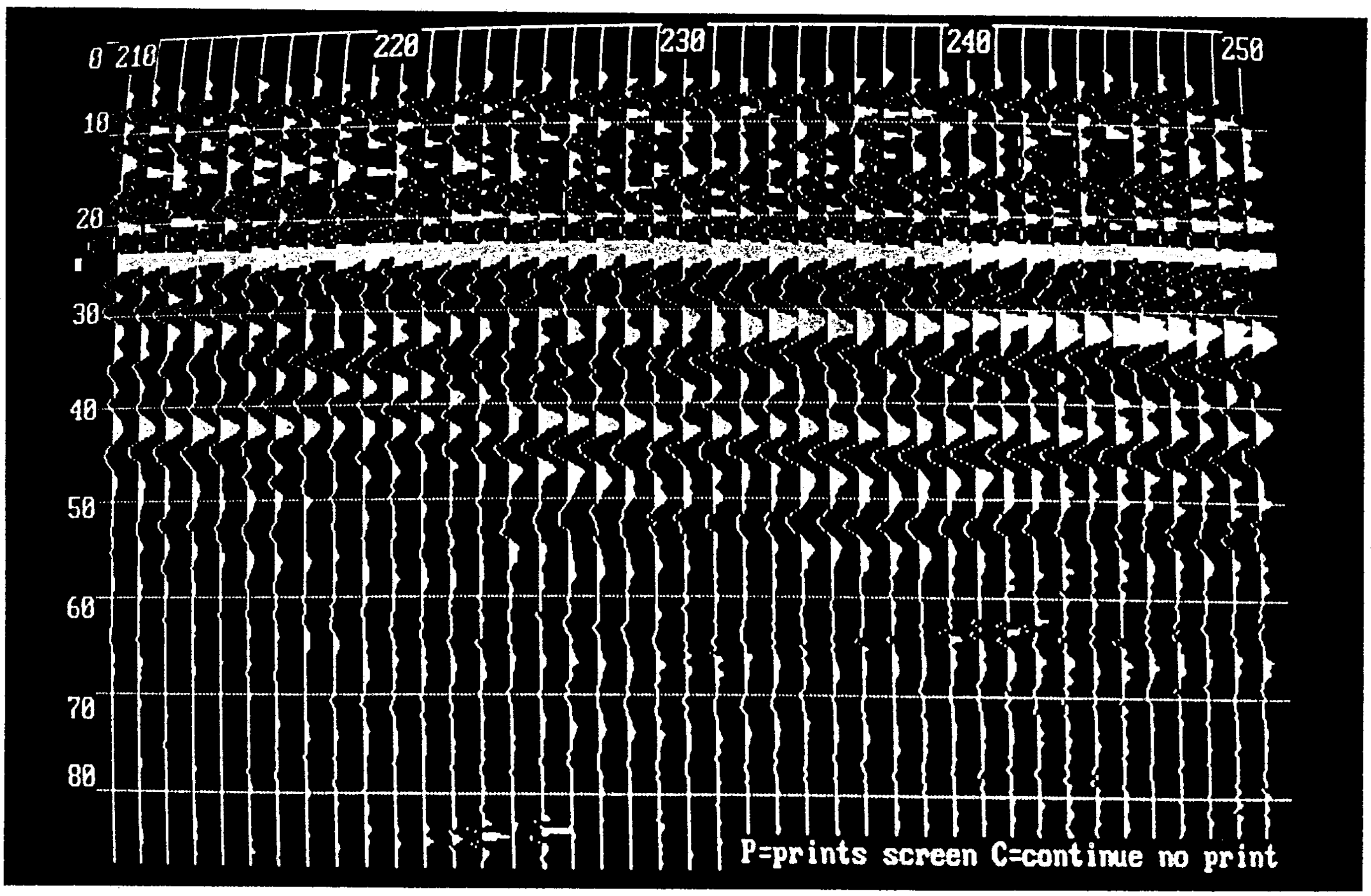
Field data can be transferred from seismic field tape onto the microcomputer in 16-bit fixed point format by any one of the following methods:

1) Parallel transfer from the DHR-2400 (I/O, Inc.) recording system's tape drive;
2) Serial transfer from a computer with a tape drive to microcomputer via RS232 interface;

   or
3) Direct transfer from tape drive to microcomputer with tape transport equipped microcomputer.

DATA DISPLAY

Seismic data can be displayed on the CRT according to user-defined parameters (start & end record numbers, start time, trace spacing, plot normalization with scan-delay time, and number of most-significant bits of trace amplitude to display). The EGA color video-interface card offers an on-screen resolution of 640x350 pixels. Option for hard copy is also available using Epson compatible graphic printer.



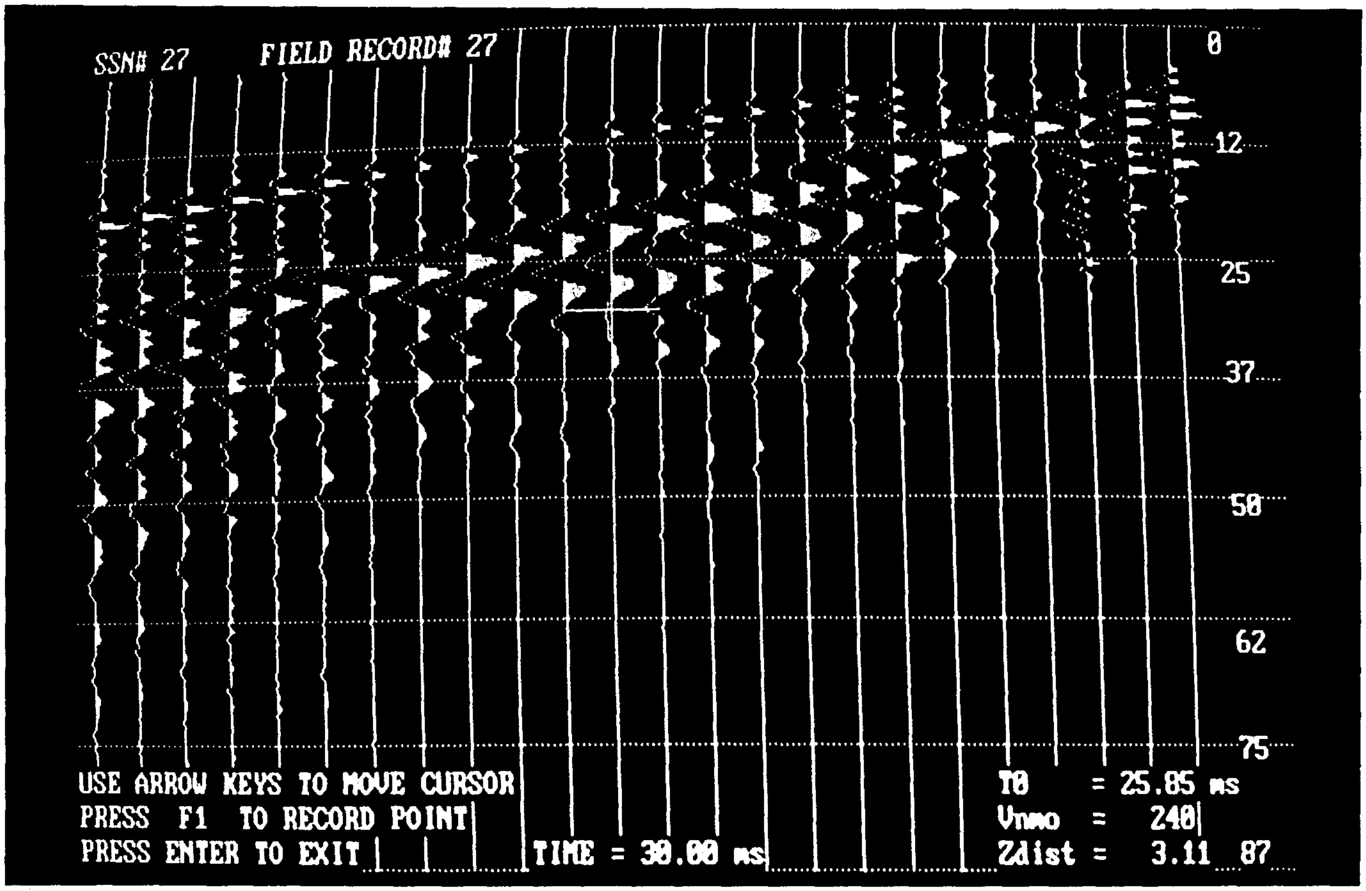
Display of final stacked section on the microcomputer CRT with 640x350 pixels resolution (EGA).

Hard copy of final stacked section generated using Epson graphic dot-matrix printer with resolution of 120x144 dots per inch.

VELOCITY ANALYSIS

Velocity Calculation from Reflection Event

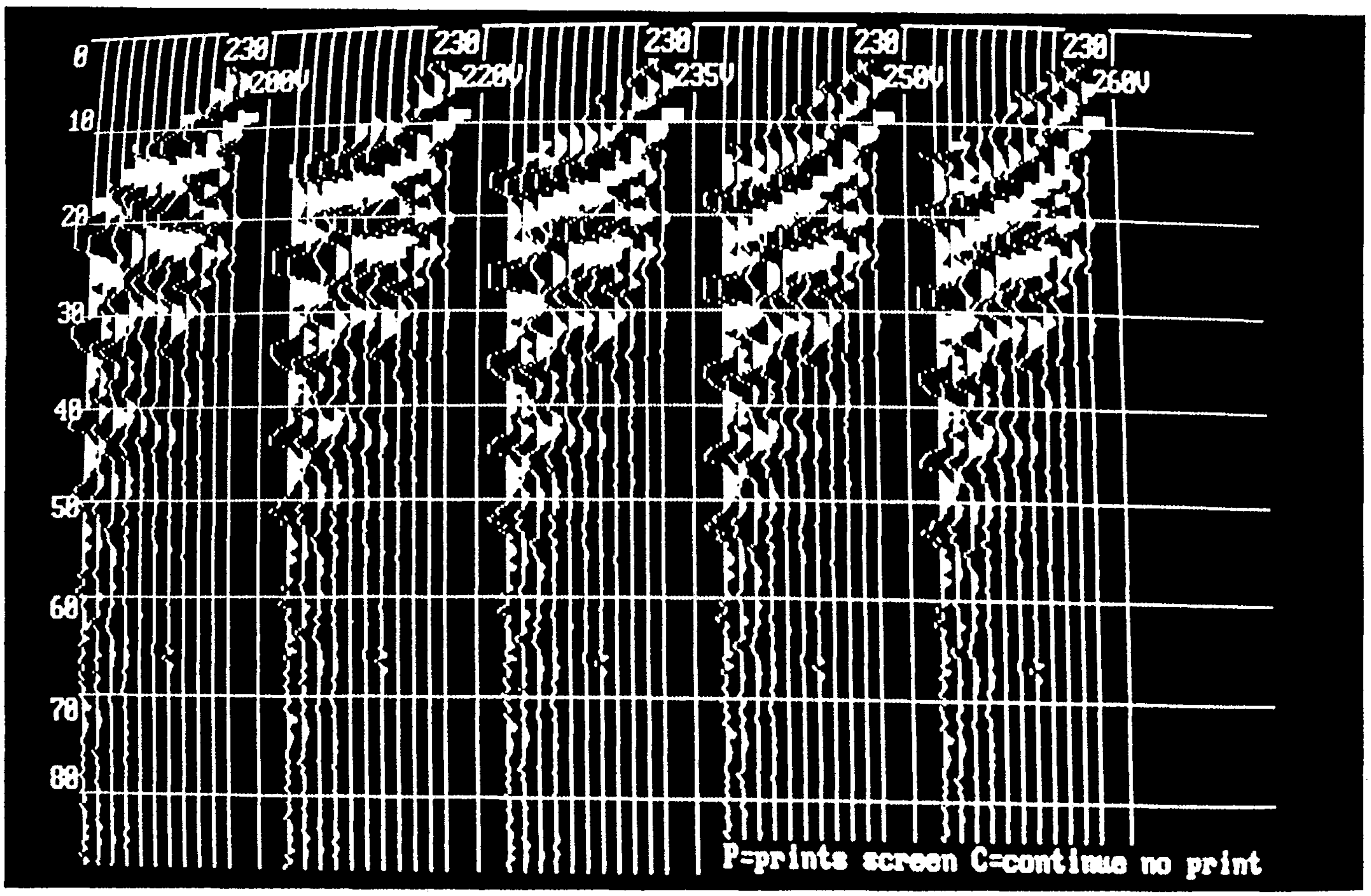
The program graphically displays a single field or CDP record on the CRT. The user can move a screen cursor along a chosen reflection event and by pressing a function key can record values of reflection time and shot-to-receiver distance. The calculated velocity is based on a least-square hyperbolic curve-fitting of the relationship between reflection time and shot-to-receiver distance.



Velocity calculation from reflection event. The screen cursor is located at trace 12 (left to right) between 25-ms and 37-ms time-lines. The calculated travel-time, average velocity, and depth of reflector are displayed at the bottom right corner of CRT.

Velocity Scan Analysis of CDP Data

This program allows the user to apply trial stacking-velocities to the same set of CDP data. Based on the input velocities, a normal moveout correction is applied to the data. The option is available for either NMO-corrected only (output as CDP-gather) or followed with CDP-stacking (output as CDP-stacked). The output velocity-scan data must be displayed using the data-display program.



Velocity scan applied to CDP-gather number 230 with trial velocities of 200, 220, 235, 250, and 260 meters/second.

CARD-CONTROL PROCESSING

Card-control processing offers flexibility to the user in the application of various procedures in any order. For example, it is possible to go directly from field data to brute stack in a single processing run, while saving selected intermediate data sets.

```
! Brute-Stack Control Deck : CDPsort, Nmo, Stack, ScaleAGC
! Save : Intermediate CDP-sorted data, Final brute-stacked data

>>START

*Inpf     FIELD.DAT
*Sort
Ptrn       .25    24      1
Pn        1      100     124 101
Shot
Sn       21            96         1    0    0    1    21 24
Sn       22            97         1    0    0    1    22 24
Sn       23            98         1    0    0    1    23 24
Sn       24            99         1    0    0    1    24 24
Snsn   < 25 47 1 >   < 100 122 1 >  1    0    0    1    23 24
Tabl
*Outf     CDPSORT.DAT

*Nmot     0.8
Velf     200      20   230
*Stak
*Scal    50
*Outf     BSTACK.DAT

>>END
```

Example of card deck organization to apply CDP sort, NMO, stack, and AGC within a single run, while saving intermediate CDP sorted and final stacked data.

Available card-control processing listed below.

Data Input To specify seismic data set to be processed.

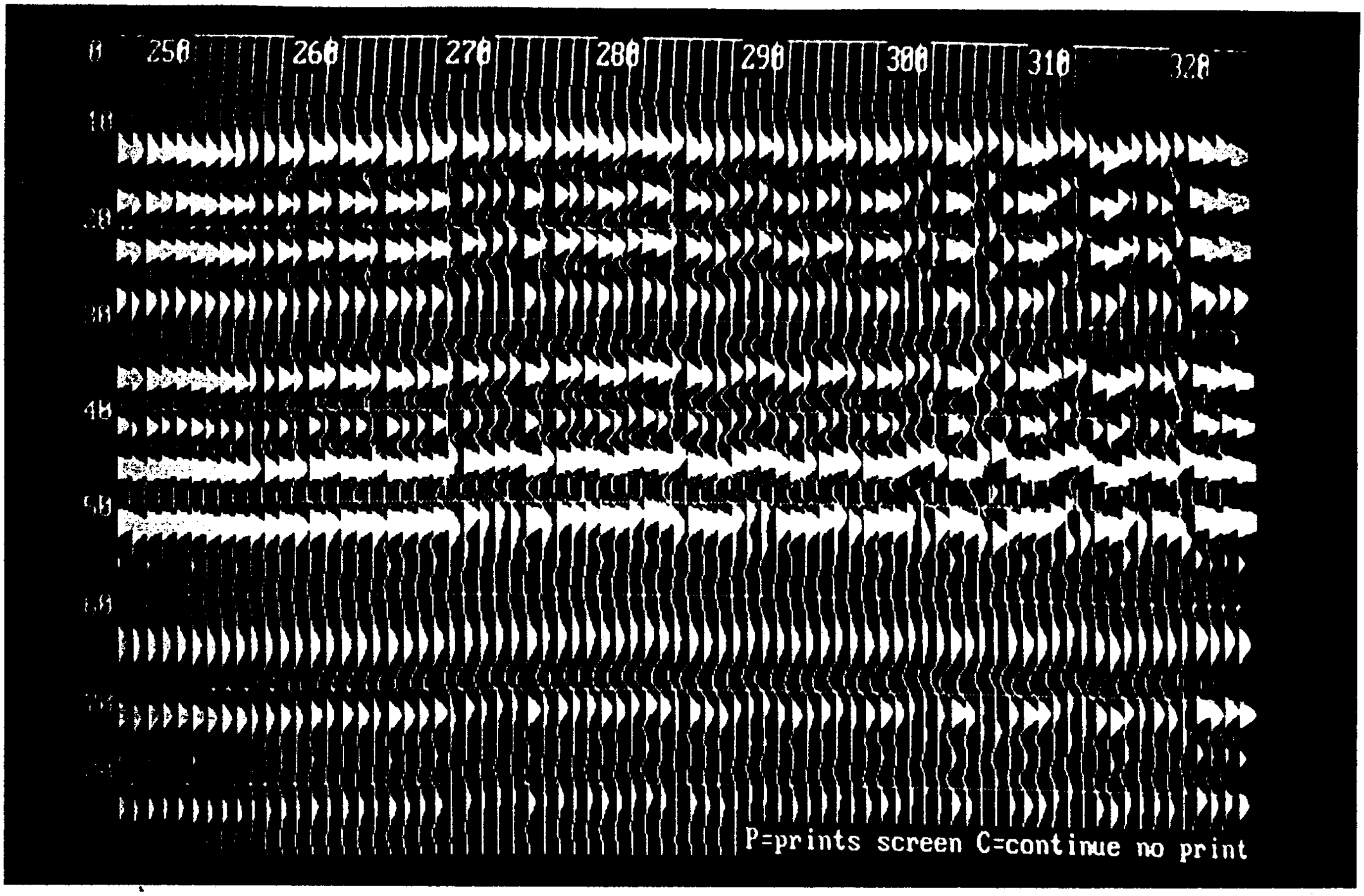
Data Output To store intermediate and/or final processed data sets.

Trace Sort To sort seismic trace based on specified trace header indices. Examples are common offset, common recording channel, common depth point, common receiver station, and common shot station.

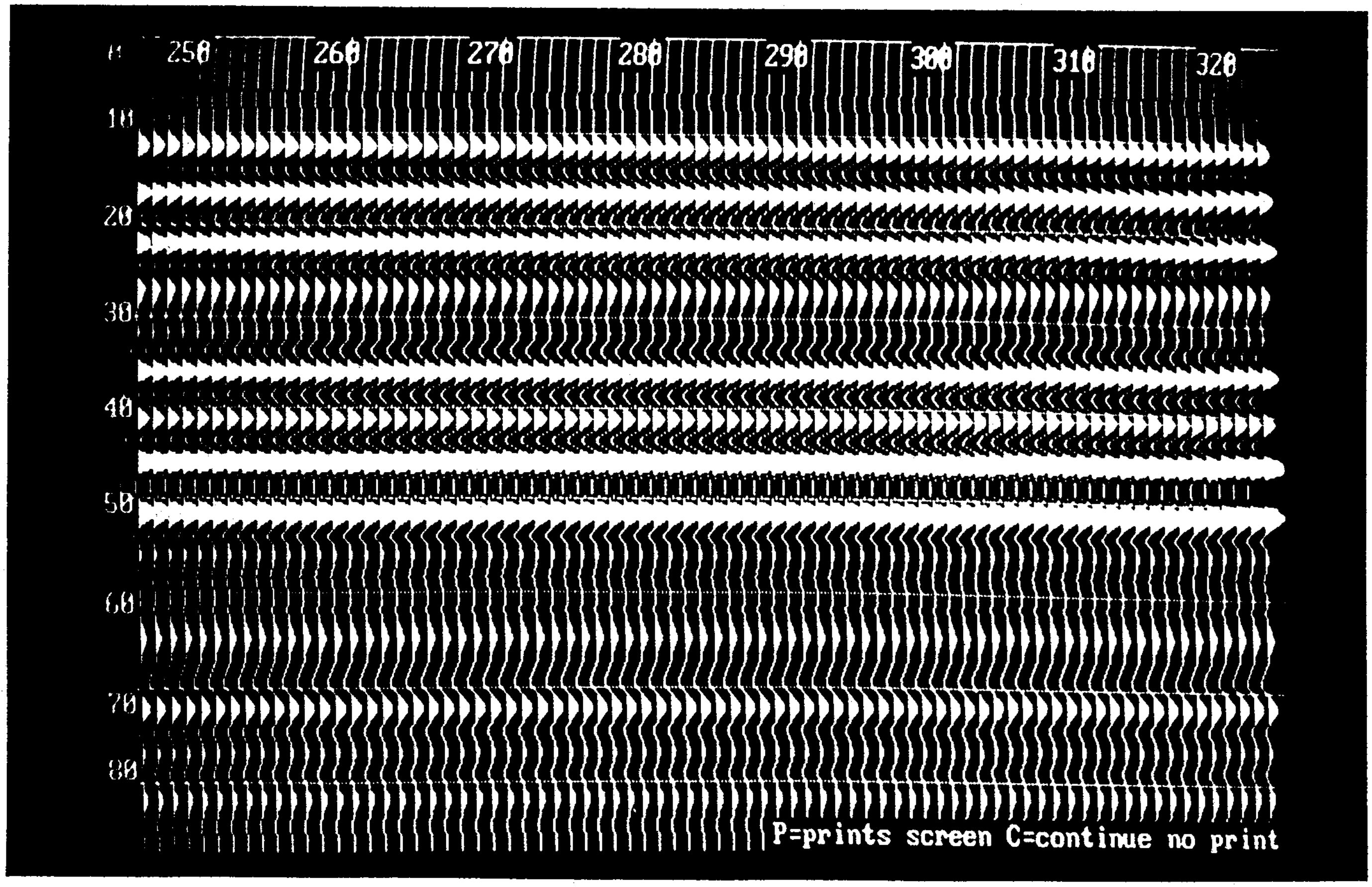
Trace Edit To edit seismic trace. Options are first arrival muting, surgical muting, and dead trace removal.

Static Correction To apply static shift to seismic trace. Options are:

1) Surface Elevation Statics: to correct shot and receiver elevations to user's defined datum based on specified average-velocity layer model,
2) Automatic Statics: to correct shot and receiver elevation for "surface consistency." The procedure is based on an iterative averaging technique along common source and receiver planes.



Synthetic CDP-stacked section *without* automatic static correction. Shot and receiver statics are randomly generated between +6 and -6 ms.



Synthetic CDP-stacked section *with* automatic static correction.

3) Residual Statics: to apply residual statics based on cross-correlation of seismic trace with the reference pilot trace.

Normal Moveout Correction To apply dynamic correction to CDP-gather to compensate for different shot-to-receiver distance. The exact NMO equation was implemented. The program also eliminates distortion caused by NMO stretching. User can specify multiple (time and space varying) velocity functions.

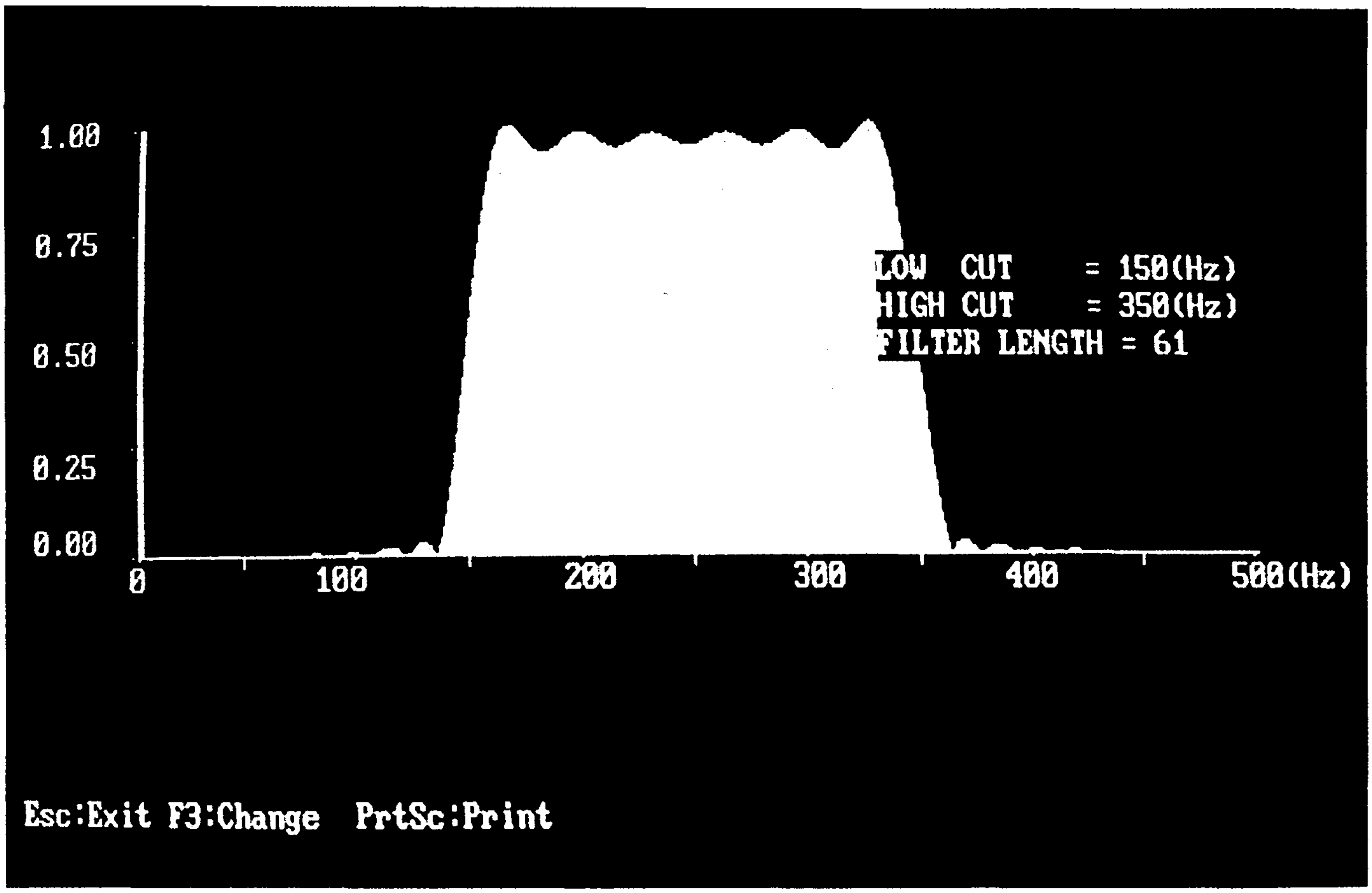
CDP Stacking To produce a single trace for each CDP by averaging the corresponding CDP-gather. With proper NMO correction previously applied, CDP stacking functions as a filter to enhance signal-to-noise ratio.

AGC Scaling To normalize the amplitude of a seismic trace based on an Automatic Gain Control procedure. It simulates feedback normalization by applying a variable gain factor to each sample based on the average amplitude of the surrounding data within the time window.

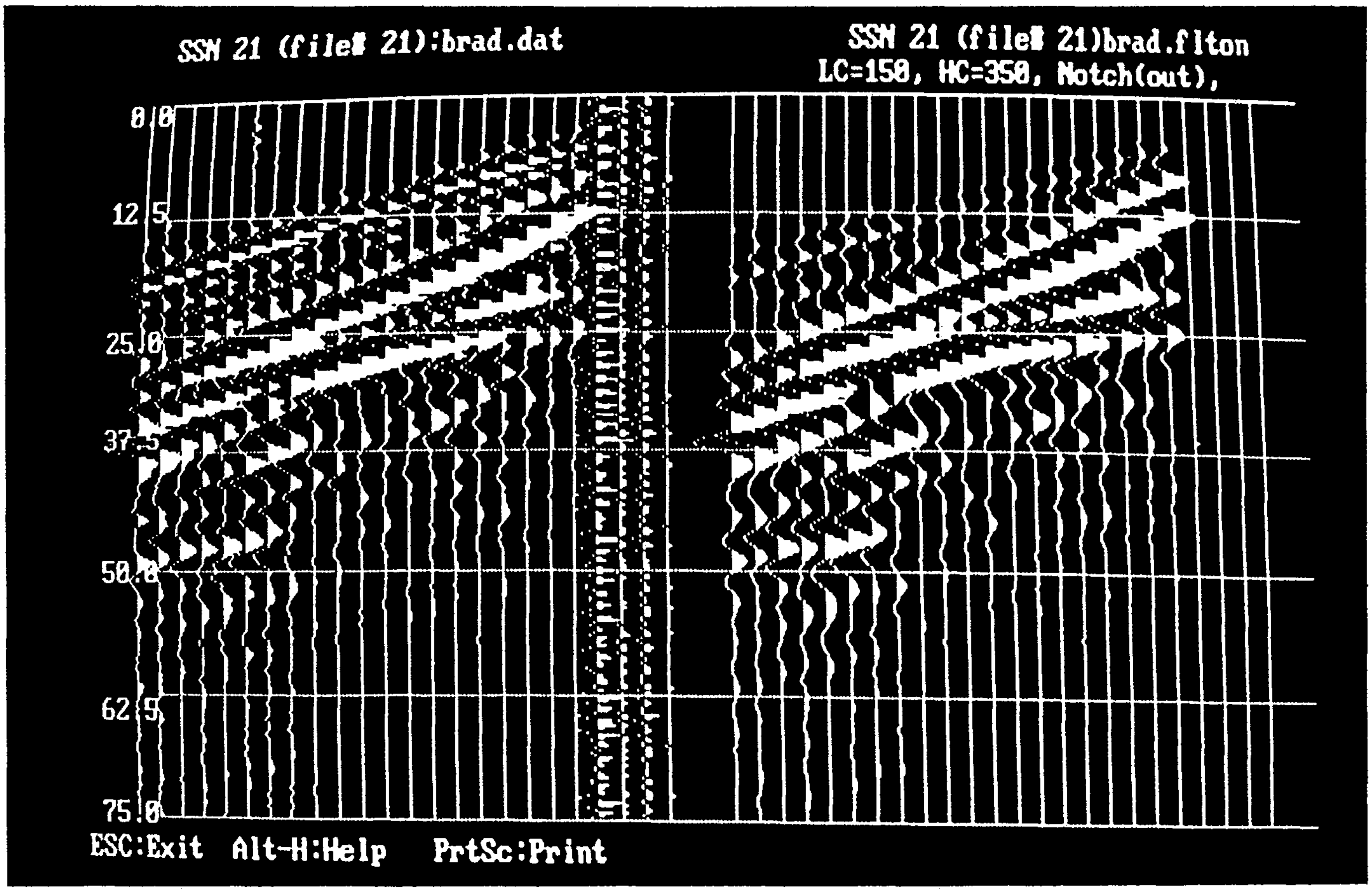
ENHANCED DATA PROCESSING

Filtering

Filtering is applied to eliminate undesired frequency components in the data. A zero-phase finite-impulse-response (FIR) filter with variable length has been implemented. Various types of filters are provided, including low-pass, high-pass, band-pass, and notch.



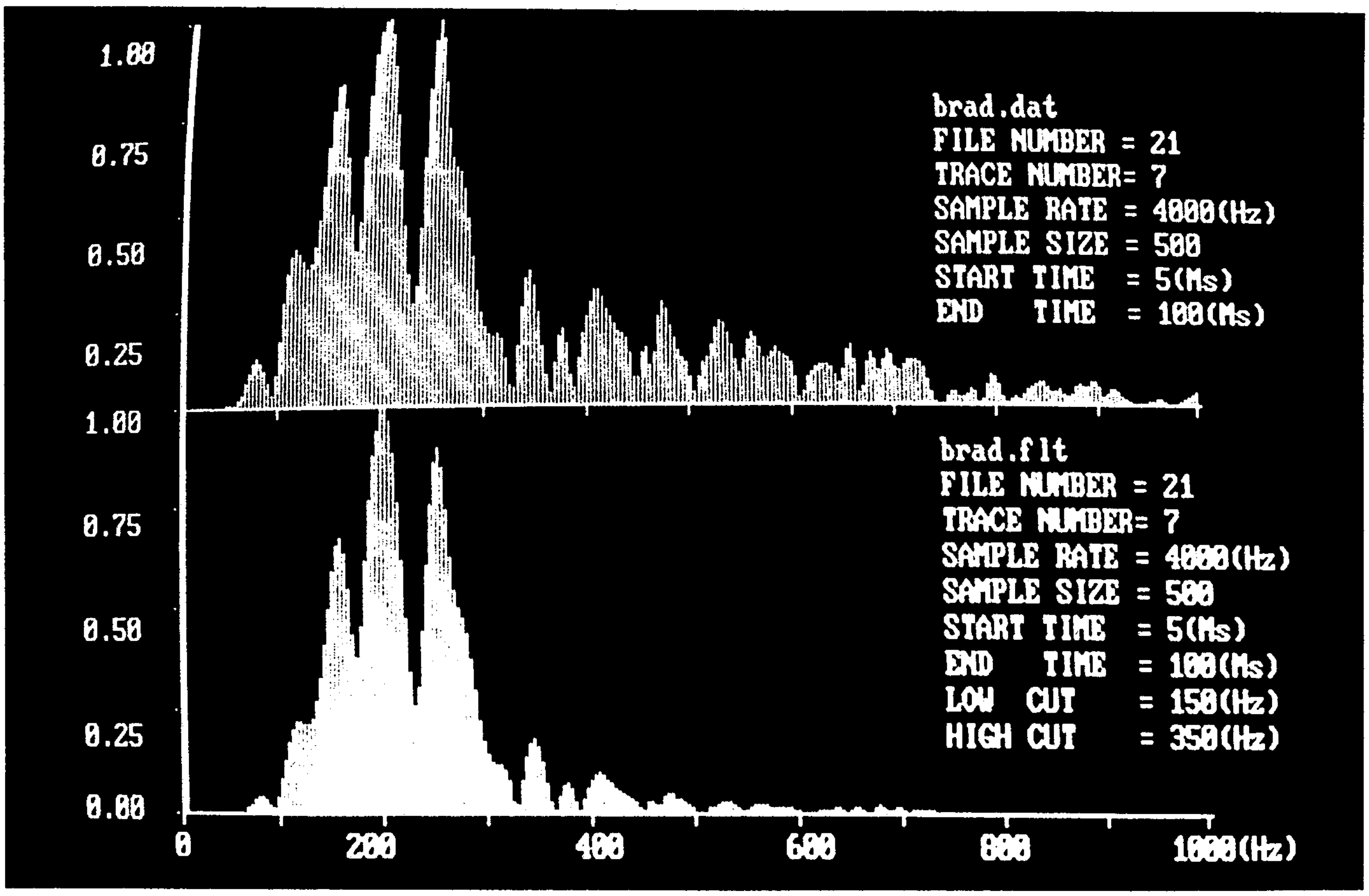
Filter characteristics of 61-point length with 150 Hz low-cut, and 350 Hz high-cut.



Comparison of unfiltered (left) and filtered (right) field data. Filter parameters are 61 points, 150 Hz low-cut, and 350 Hz high-cut.

Spectral Analysis

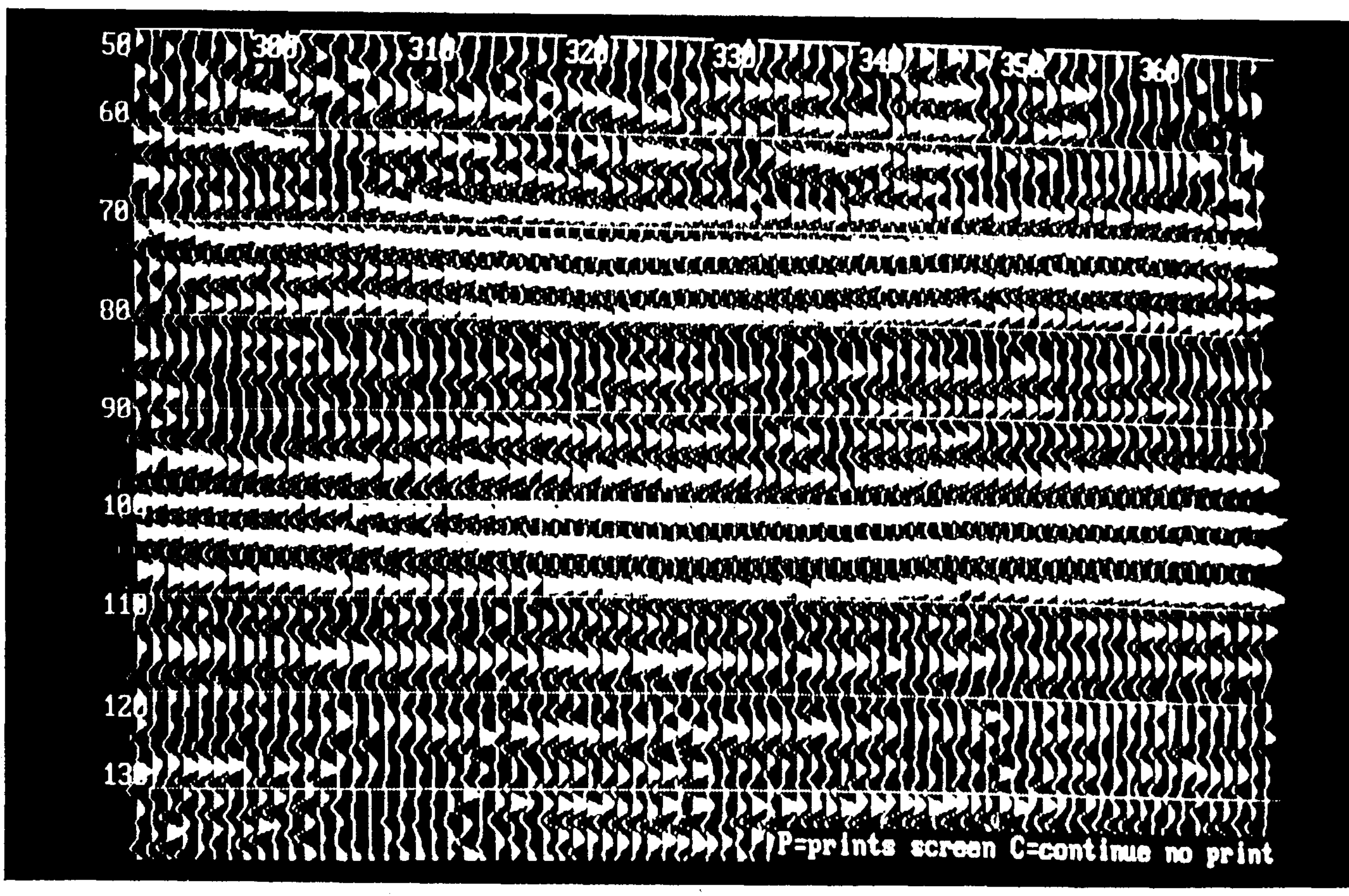
Power spectra of data before and after filtering within user-specified time window can be displayed.



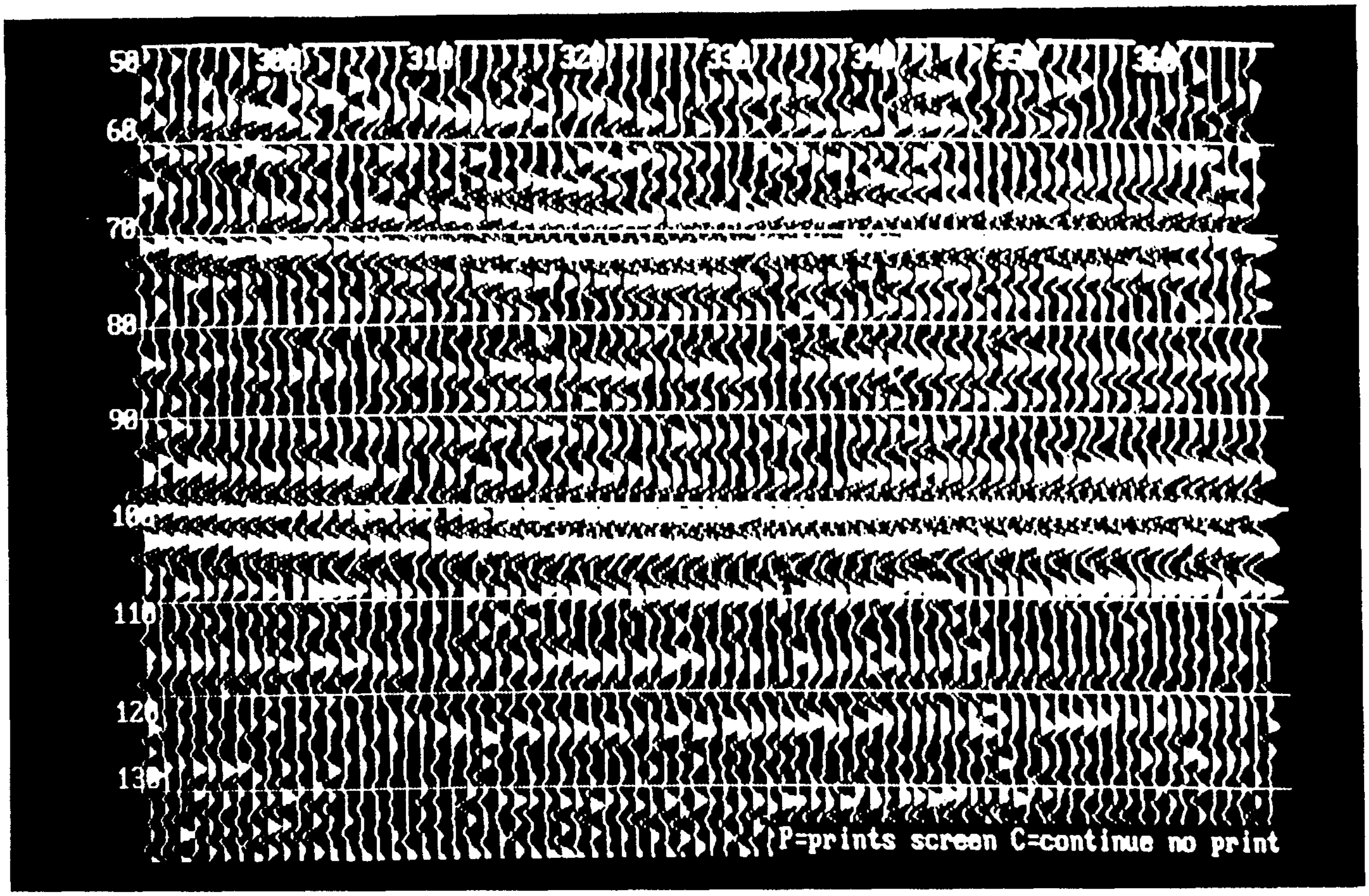
Comparison of spectra of unfiltered (top) and filtered (bottom) seismic trace.

Deconvolution

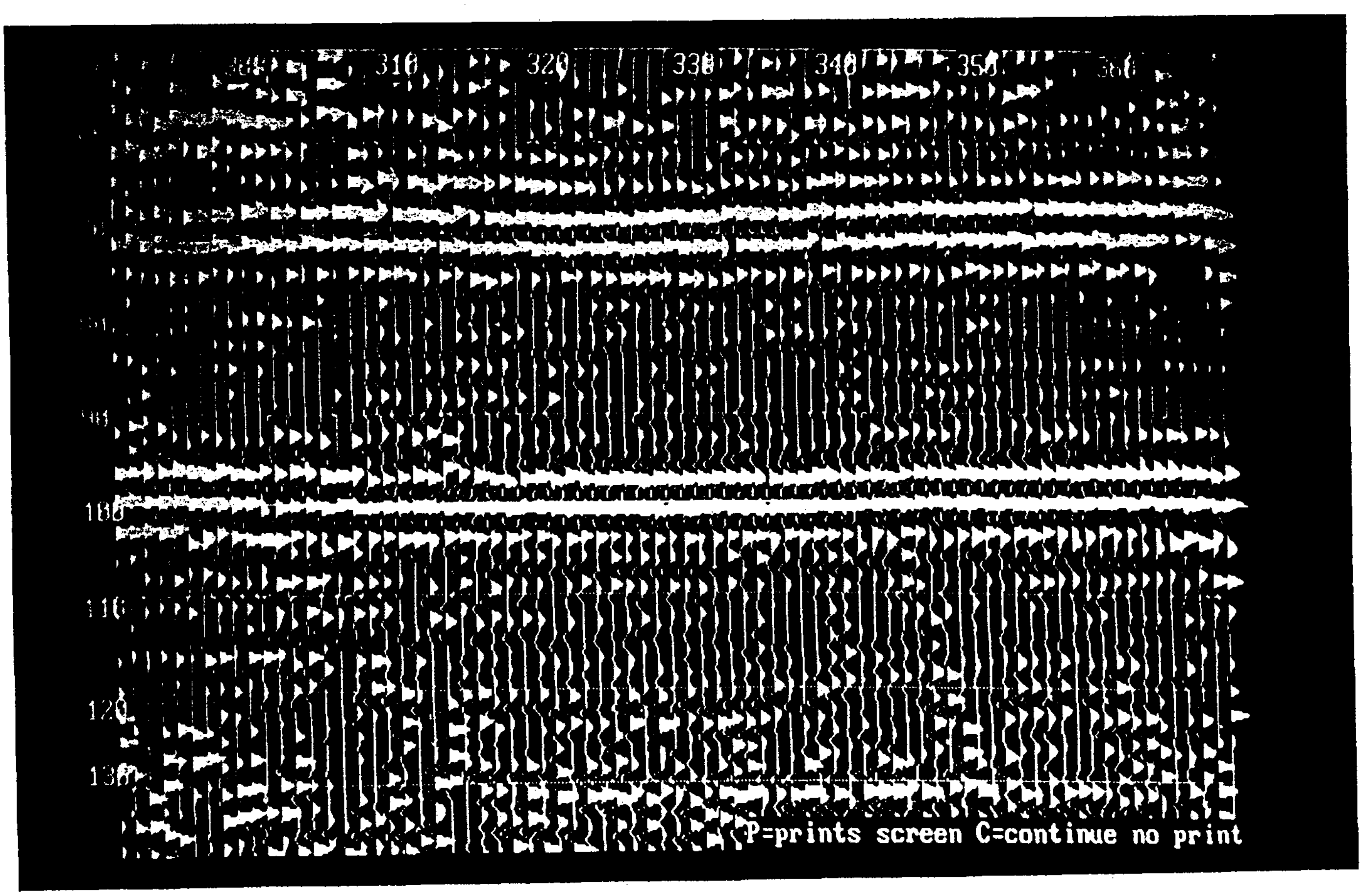
Deconvolution is applied to remove multiples or reverberated signals from the data. It is based on the least-square predictive deconvolution method. Options are available for spiking and 2nd-zero-crossing predictive. The operating parameters include percentage spectral whitening, time window for auto-correlation, and filter length.



Stacked data *before* applying deconvolution.



Stacked data *after* applying *2nd-zero-crossing predictive* deconvolution with 4% whitening and auto-correlated window from 20 to 150 ms.



Stacked data after applying spiking deconvolution with 4% whitening and auto-correlated window from 20 to 150 ms.

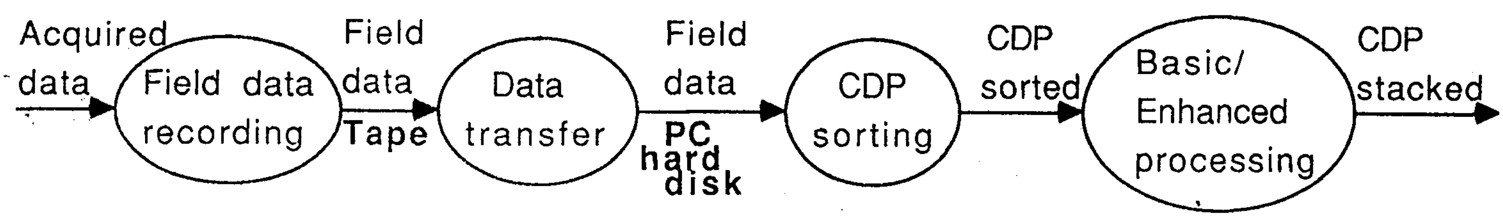
PROCESSING SPEED*

Procedure	Time	Data Size/Parameters
Parallel data transfer	17 sec	500 samples
Serial data transfer	25 sec	500 samples
CDP sorting	5 min	100 shotpoints (24 channels/shotpoint, 500 samples/channel)
NMO correction and CDP stacking	14 min	100 shotpoints (24 channels/shotpoint, 500 samples/channel)
Automatic (surface consistent) statics	16 min	120 CDPs (12 folds/CDP, 160-sample correlated window, 12-sample static limit)
Filtering	1.5 sec	500 samples (filter length = 31 points)
Deconvolution	6 sec	500 samples (filter length = 31 points)

**IBM-PC/AT compatible microcomputer (6 MHz) with 80287 math co-processor*

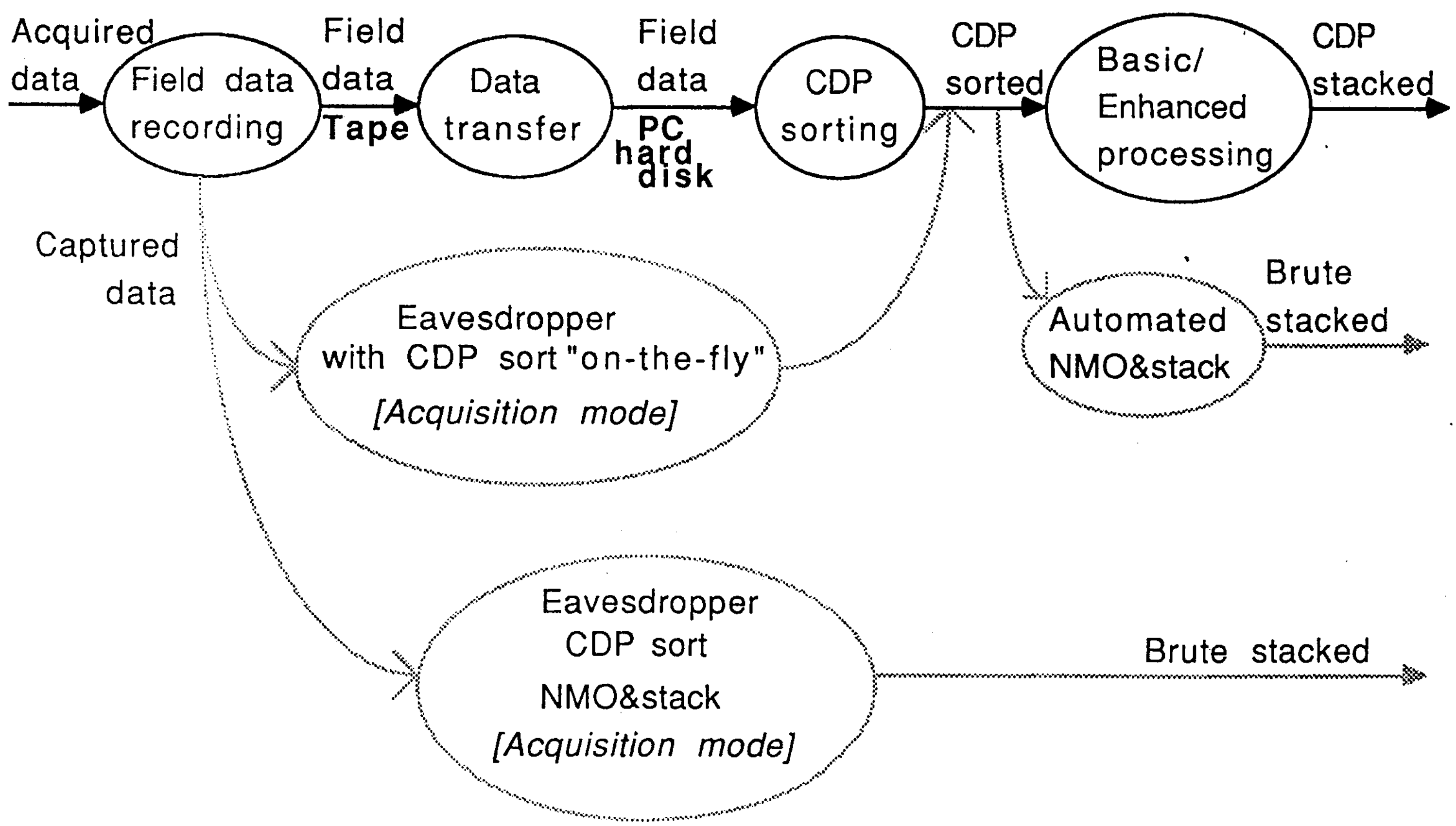
CONCLUSIONS & FUTURE DEVELOPMENT

As previously presented, seismic data processing is possible with a microcomputer. Processing techniques developed here include both basic and some enhancement procedures. Additional features under development include migration, velocity spectra, color display analysis, automatic trace editing, and automatic normal moveout correction. All the previously described processing techniques are applied *after* data acquisition, and can be summarized in the data flow diagram below.



Data flow diagram of CDP processing *after* data acquisition

Another approach is under consideration where basic processing could be accomplished *during* data-acquisition mode. A hardware/software design, called the *Eavesdropper (Bennett & Chung, Geophysics: The Leading Edge, July 1986)*, has been implemented to capture field data from the DHR-2400 (I/O, Inc.) recording system during data acquisition. The plan is to improve its capability so that it is able to do CDP sorting "on-the-fly," after which brute stack could be produced, provided that an automated normal moveout correction with CDP stacking is available.



Data flow diagram of CDP processing *during* and *after* data acquisition