Standard Data Format Utilities
User's Guide

Version B.02.01

Registration Card

Make sure you send in the product registration card located in the red brochure
included in your SDF Utilities shipping carton. This assures that you will hear
about future SDF Utilities product and service updates. If the brochure has been
misplaced, simply mail or fax the following information to the indicated
address/fax number.

- Your Name:
- Position:
- Company Name:
- Division:
- Mail Stop:
- Street:
- City:
- State or Province:
- Postal Code:
- Country:
- Telephone: (   )-
- Instrument Model:
- Serial Number:

Mail to:

> ATTN: Quality and Productivity MS230
> Hewlett-Packard Company
> Lake Stevens Instrument Division
> 8600 Soper Hill Rd.
> Everett, WA 98205-1298

Or FAX:

> (206) 335-2828
> No cover sheet is required.
> Quality and Productivity Department MS230

TABLE OF CONTENTS

CHAPTER 5: Viewdata

CHAPTER 6: Using SDF Data with C Programs

General Information

Introduction

Standard Data Format Utilities is a group of MS-DOS® programs for doing the following things (see figure 1-1, on the following page):

- Converting files between LIF format and DOS format.
- Sharing data between HP DSA analyzers.
- Displaying analyzer data on your PC.
- Reading data from files into a program.

(MS-DOS is a U.S. registered trademark of Microsoft Corporation)
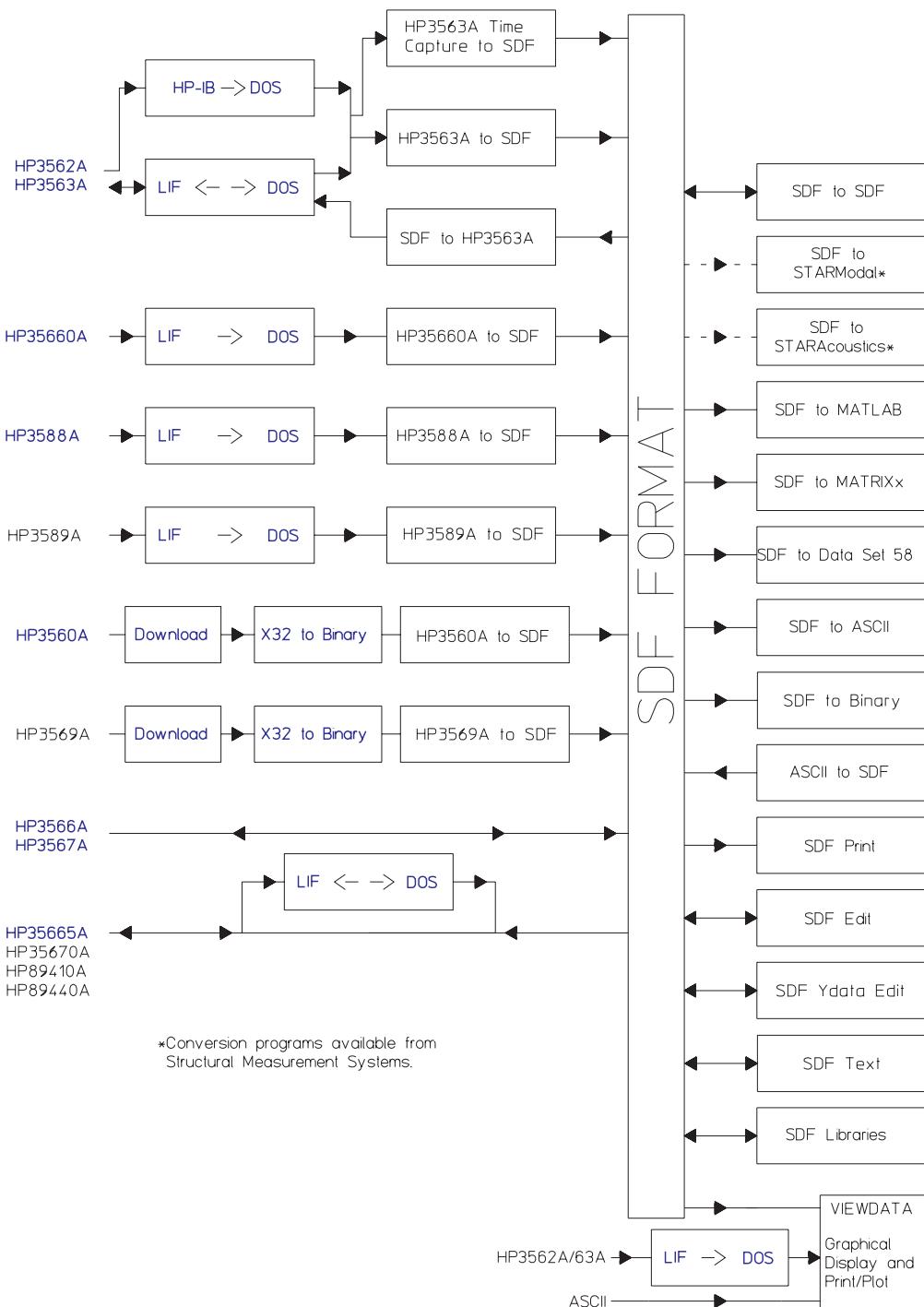
Figure 1-1. Data Exchange Capabilities

Notice in figure 1-1 that the data from some analyzers, such as the HP 3563A, must be converted first from LIF to DOS, then to SDF format. For some other analyzers, such as the HP 3566A, no conversions are necessary—these analysers save data in SDF format directly.

The Standard Data Format Utilities support saved data from the following DSA analyzers:

- HP 35665A (to and from)

- HP 35670A (to and from)

- HP 3562A (to and from)

- HP 3563A (to and from)

- HP 3566A/3567A (to and from)

- HP 35660A (from only)

- HP 3560A (from only)

- HP 3569A (from only)

- HP 3588A (from only)

- HP 3589A (from only)

- HP 89410A/HP89440A (to and from)

- HP 3587S (time capture files only)

There are some limitations on data exchange. For example, order tracking data from an HP 3567A cannot be post-processed in an HP 3563A because the HP 3563A does not do order tracking measurements.

The Standard Data Format Utilities also allow you to translate data files to the following formats compatible with third party software:

- PC-MATLAB, a trademark of The MathWorks, Inc., is a software package for general digital signal processing and filtering.

- MATRIXx, a product of Integrated Systems Inc., is a software package for control system analysis.

- Data Set 58 is the universal ASCII format for mechanical test.

- ASCII is a versatile format for spreadsheets and other general software.

Programs that convert from Standard Data Format to SMS formats are available from SMS for use with the following application software:

- STARModal, a trademark of SMS, is a software package for doing modal analysis of structures, including structural modification and force/response analysis.

- STARAcoustics, a trademark of SMS, is an acoustics software package with 1/3 octave, sound intensity, and sound power analysis.

Converting Files Between LIF Format and DOS Format

LIF (Logical Interchange Format) is the Hewlett-Packard standard disk format that may be used to exchange files among various HP computer systems and instruments. The LIF programs in the Standard Data Format Utilities make it possible for personal computer users to translate LIF files into DOS file format, and DOS files into LIF format. The utilities support file conversion for both internal drives and external drives connected on the HP-IB. These utilities can also format and initialize LIF disks on external HP drives.

README.TXT

This text file, shipped on the SDF UTILITIES disk 1, provides additional information about SDF UTILITIES. Read and/or print the contents of README.TXT as you do with an ASCII text file.

Data Sharing Between Analyzers

These utilities make it possible to take data with one analyzer and recall it into another analyzer for post-processing. (Chapter 4 describes these utilities in detail.) For example, you can make a frequency response measurement with an HP 3560A, transfer the data to your PC, convert it to SDF format, then recall it into an HP 35665A for curve fitting.
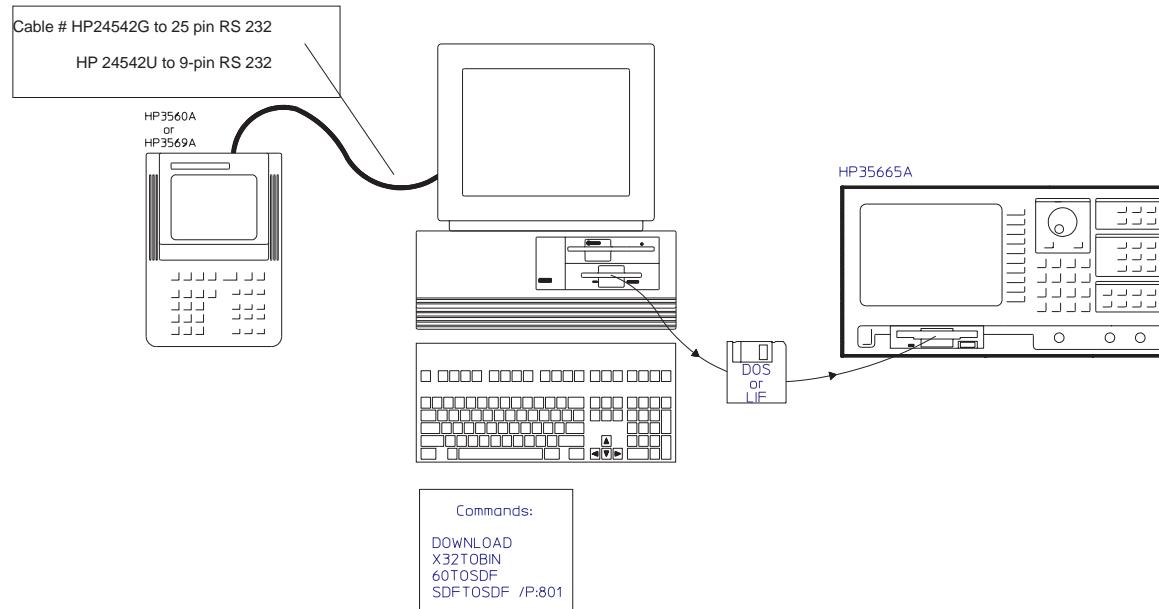
Figure 1-2. HP 3560A to HP 35665A

To connect the HP 3560A or HP 3560A to the PC: use Cable #HP 24542G to 25-pin RS-232 or Cable #HP 24542U to 9-pin RS-232. For more information on the cables required to connect the HP 3560A to the PC, refer to the chapter ''Transferring Data to a Personal Computer'' in the *HP 3560A Quick Start Guide*.

Some analyzers only accept data with 401 frequency lines, others accept only 801 frequency lines, and others allow you to specify more or fewer frequency lines. The SDFTOSDF utility (see chapter 4) allows you to convert data from one number of frequency lines to another. For example, if your HP 3560A or HP 3569A measurement was done with 1601 lines, you must convert it to 801 lines before you can curve fit it in an HP 3563A, using the command SDFTOSDF /P:801. (Although the HP 35665A can only create 401 line data, it can display and curve fit data with 801 lines.)

You can compare old data from an HP 3562A to new data from an HP 35665A, as shown in figure 1-3. You can convert the HP 3562A data from LIF to DOS, then from HP 3562A format into SDF, then recall it into an HP 35665A for comparison. Or you can convert HP 35665A data from SDF format to HP 3562A format, then convert it from LIF to DOS, then recall it into an HP 3562A for comparison.
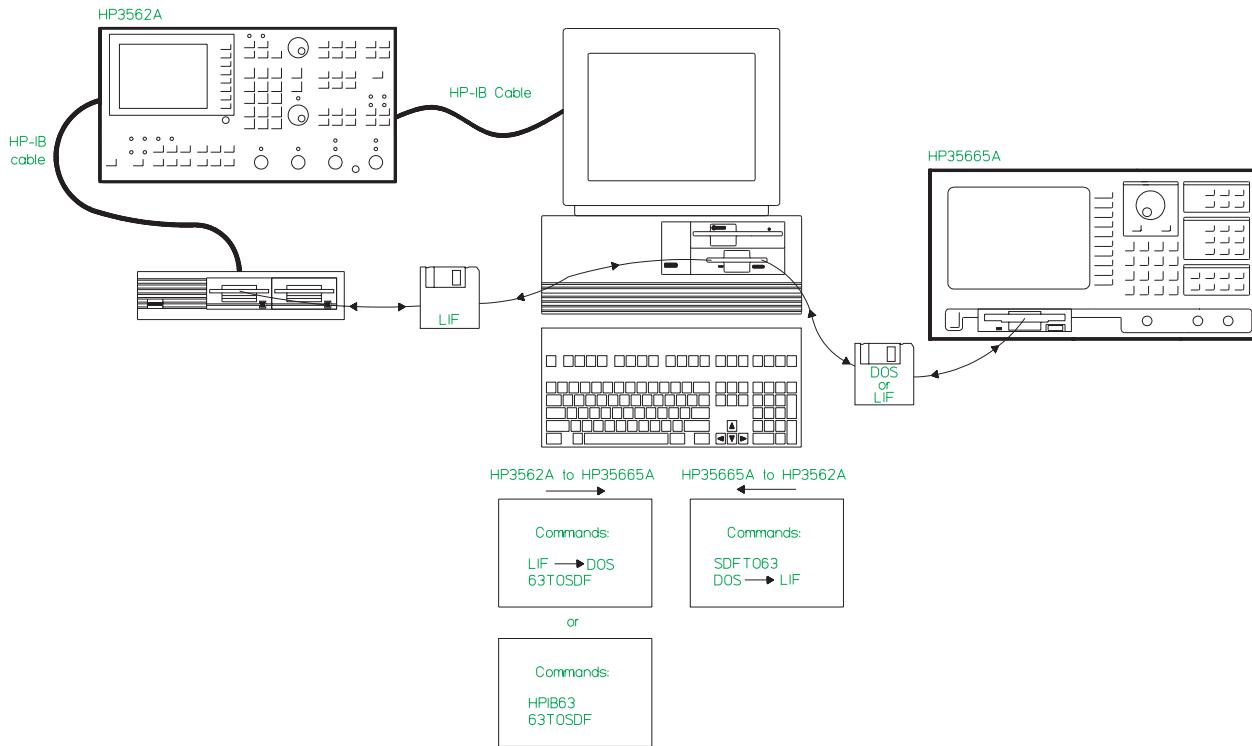
Figure 1-3. HP 3562A to HP 35665A

Batch Files

Appendix C contains several example batch files. These are ''automated'' procedures that combine two or three utilities. For example, ''63_SDF.BAT'' performs the LIF-to-DOS conversion and the HP 3563A-to-SDF conversion on a group of files. You can use the existing batch files, or you can copy and modify them for your particular needs.

Displaying Analyzer Data on a Personal Computer

The Viewdata utility allows you to display data graphically on your PC. You can display up to three traces of data from the same or different analyzers. For example, you can display a frequency response from an HP 3563A on trace A, a frequency response from an HP 35665A on trace B, and a frequency response from an HP 3566A/3567A on trace C. Chapter 5 describes the Viewdata utility in detail.

Note   You must convert HP 3563A/62A data from LIF to DOS before reading it into Viewdata. See chapter 3.

The utility includes marker, scale, zoom, and scroll functions. You can also plot or print the display to HP-GL plotters or PCL printers (for example, the HP LaserJet or DeskJet).

Figure 1-4. Viewdata

Reading Data from Files into a Program

The functions in the SDF Libraries allow you to write your own C programs accessing SDF data via function calls. Chapter 6 describes libraries and includes a sample program.

Minimum Requirements

To run the utilities, you need the following:

- An IBM PC compatible and MS-DOS 2.1 (or greater). You can run the software from a high-density flexible disk drive or from your hard disk drive (a minimum of two drives are required).
- At least 256 kilobytes (K) of memory, 384K of memory is recommended.

Installation

To install the Standard Data Format Utilities, copy all files on the two disks to any directory on your PC. Or, if you do not have a hard disk, you can insert the disk in an internal drive and type commands from there.

Note   If you are running the utilities from a floppy drive, you cannot address LIF files in that drive.

The following files are on the disks:

60_ML.BAT	89TOSDF.EXE	SDFPRINT.EXE
60_MX.BAT	ASCTOSDF.EXE	SDFTEXT.EXE
60_PLOT.BAT	DEFECT.TXT	SDFTO58.EXE
60_SDF.BAT	DFDATA63.EXE	SDFTO63.EXE
60TOSDF.EXE	DFHDR63.EXE	SDFTOASC.EXE
63_ML.BAT	DOWNLOAD.EXE	SDFTOBIN.EXE
63_MX.BAT	HEADER63.EXE	SDFTOML.EXE
63_PLOT.BAT	HPIB63.EXE	SDFTOMX.EXE
63_SDF.BAT	IB63_ML.BAT	SDFTOSDF.EXE
63TCSDF.EXE	IB63_MX.BAT	SDFYDATA.EXE
63TOSDF.EXE	IB63_PLT.BAT	SETUP63.EXE
660TOSDF.EXE	IB63_SDF.BAT	SETUP88.EXE
69_ML.BAT	LIF.EXE	SETUP89.EXE
69_MX.BAT	LIFDIAG.EXE	SOFTCOPY.EXE
69_PLOT.BAT	README.TXT	SYNTH63.EXE
69_SDF.BAT	REPEAT.EXE	USAGE.TXT
69_TOSDF.EXE	REVID	VIEWDATA.EXE
88TOSDF.EXE	SDFEDIT.EXE	X32TOBIN.EXE

If you want to use the SDF libraries, you must also copy the following files from the directory SDFLIB:

README.TXT	SDF.C	SDF.MAK
SDF_BASI.H	SDF_COMM.H	SDF_UTIL.H
SDFADD.C	SDFDATA.C	SDFMPY.C
SDFUTIL.LIB		

What is Standard Data Format?

Introduction

Standard Data Format, SDF, is a data format that allows measurement and analysis data to be shared among several HP analyzers (see chapter 1 for a list of the supported analyzers). See appendix B for a complete description of each field in the SDF file.

SDF supports analyzers with a very broad range of functionality. Not all the supported analyzers use SDF's full capabilities. The rest of this chapter describes some of these differences. Use the SDFTOSDF utility (described in chapter 4) to extract from an SDF file the information of interest.

Measurement (Data) Results

Some analyzers save a complete set of measurement results. Others save an individual trace.

For example, when you save data from an HP 3566A/67A frequency response measurement, the file includes five measurement results (or measurement data): FRF, coherence, input power, output power, and cross power. When you save data from an HP 35665A FFT measurement, the file includes only the measurement data displayed in the active trace. In the SDF file, these are called ''data results.''
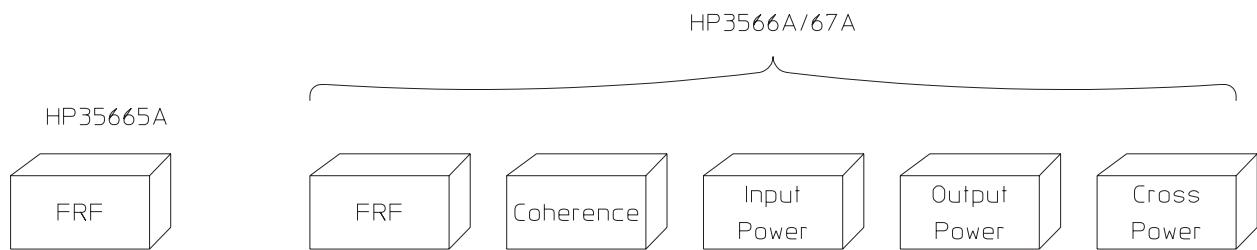
Figure 2-1. Data Results

If you want to recall FRF data from an HP 3566A/67A into an HP 35665A, use SDFTOSDF /D to select the measurement data of interest from the HP 3566A/67A data. See the SDFTOSDF command in chapter 4 for more information.

The HP 35665A can read HP 3566A/67A data directly into the instrument, but it can read only the first trace in the file with:

- 1st data result
- 1st row
- 1st column

From the HP 3566A/67A, the save trace command can be used to save a specific trace to disk.

Number of Input Channels (Rows)

Another difference between SDF files is the number of input channels. For example, the HP 3566A and HP 3567A are multi-channel analyzers with up to 16 input channels, and the other supported analyzers have only one or two input channels. In the SDF file, these are called ''rows.''
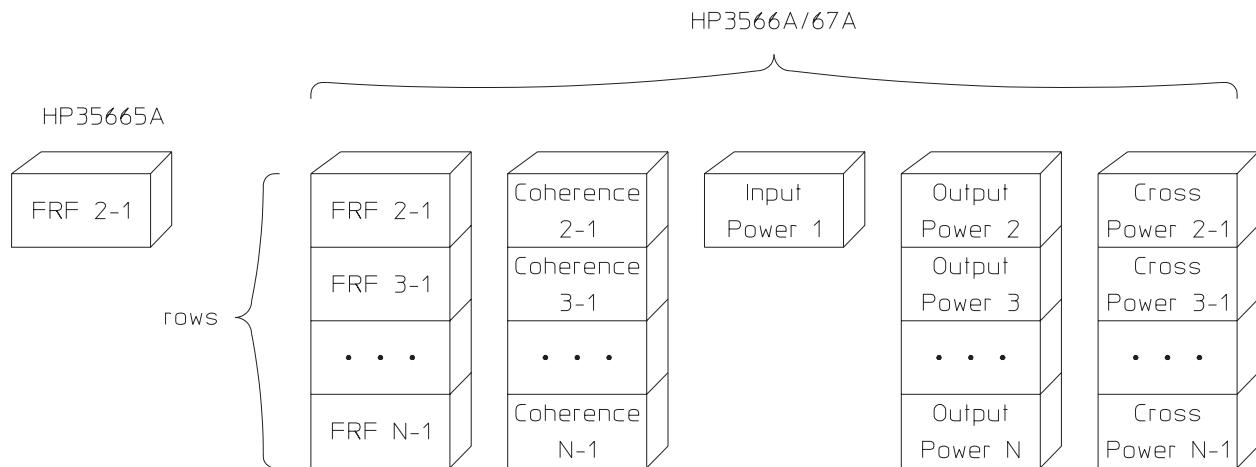
Figure 2-2. Rows

If you want to recall FRF data from an HP 3566A/67A into an HP 35665A, use SDFTOSDF /R to select the HP 3566A/67A channel combination (row) of interest. See the SDFTOSDF command in chapter 4 for more information.

Waterfalls and Maps (Scans)

Some of the supported analyzers allow you to save waterfalls or maps—a number of spectra. These may be the waterfall steps for an HP 35665A measurement or map lines for an HP 3566A/67A rpm spectral map, for example. In the SDF file, these are called ''scans.''
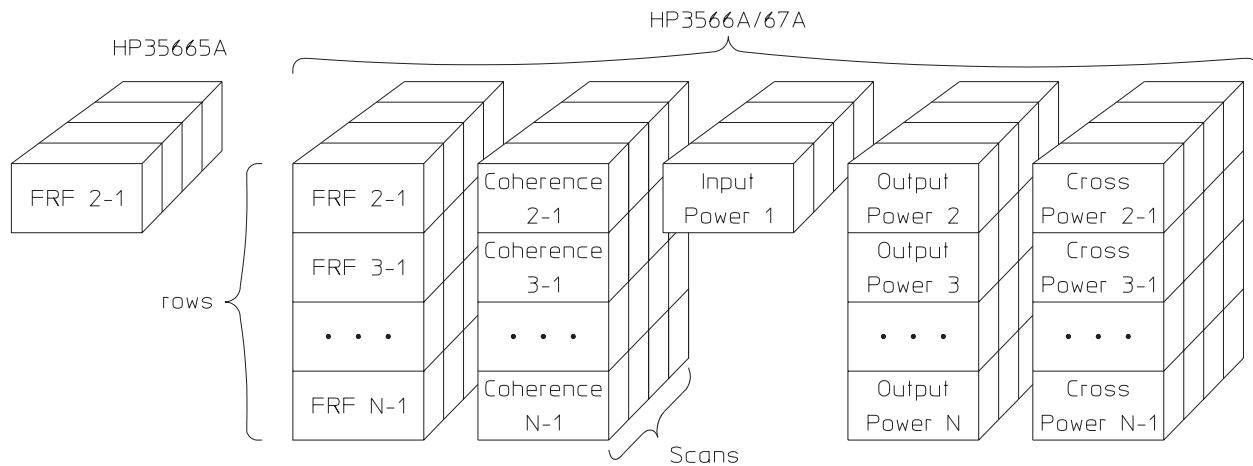
Figure 2-3. Waterfall or Map Scans

Use SDFTOSDF /S to select the scan (map line or waterfall step) of interest. See the SDFTOSDF command in chapter 4 for more information.

Time Capture (Scans)

Some of the supported analyzers allow you to capture long streams of raw time data and then later replay the data into a measurement in the instrument. The HP 3566A/3567A, HP 35665A, HP 35670A, HP 89410A, HP 89440A, and HP 3587S support this capability. The time data in the file is broken up into scans in the same manner as with waterfalls and maps (see the preceding section). The scan size is determined by the instrument and is generally related to the number of time points the instrument uses to generate an FFT in its current measurement.
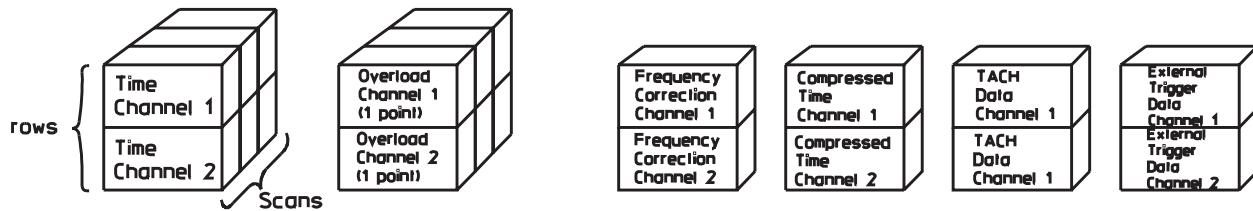
Figure 2-4. Time Capture Scans

Time Channel and Frequency Correction are generated by all instruments. Overloads are generated by the HP 3566A/3567A only. Compressed Time, Tach Data, and External Trigger Data are generated by the HP 3566A/3567A and the HP 35670A.

See Appendix B for more information on Time Capture Files.

File System Conversion (LIF/DOS)

Description

LIF (Logical Interchange Format) is the Hewlett-Packard standard disk format that may be used to exchange files among various HP computer systems and instruments. The LIF programs in the Standard Data Format Utilities make it possible for personal computer users to translate LIF files into DOS file format, and DOS files into LIF format. The utilities support file conversion for both internal drives and external drives (connected on the HP-IB). These utilities can also format and initialize LIF disks on external HP drives.

The LIF utilities consist of the following commands:

LIF   Main program for LIF access. The first parameter specifies the function (e.g., SCAN or CHK). Each function has its own options. These options are documented later in this chapter.

LIF SCAN   Checks the PC for internal floppy drives and for HP-IB interface boards and then looks for disk drives (capable of reading LIF disks) connected to HP-IB.

LIF CHK   Verifies that the disk inserted in a designated drive is fully readable by the LIF Utilities. (To abort, type: CTRL-C.)

LIF LS   Lists the contents of the LIF directory.

LIF CP   Copies a LIF file to a DOS file, a DOS file to a LIF file, or a LIF file to another LIF file.

LIF RM   Deletes a specified file from the LIF directory.

LIF INIT   Optionally formats and initializes a new directory on a LIF disk. (To abort, type: CTRL-C.)

LIFDIAG   Diagnostic utility useful for performing fast LIF to LIF full disk backups and for recovering data from damaged/corrupted disks.

Hardware and Software Requirements

Use of these utilities requires an IBM PC compatible with the MS-DOS 2.1 operating system (or greater). If you wish to translate or format LIF files located on an external drive you must have an SS/80 drive connected to your PC with an HP-IB card. The LIF utilities recognize all HP SS/80 drives including the HP 9122, HP 9133 (D,H,L), HP 9127A and HP 9153A. Older disk drives (AMIGO command set) such as the HP 9121, HP 9133 (V, X, or XV) will not work. (SS/80 disks are a subset of HP CS/80 disks.) There are no requirements for internal disk drives.

The HP 88500A and HP 82335 HP-IB cards, as well as the National GPIB PCII and AT-GPIB cards are recognized. Make sure that your HP-IB (or GPIB) is set to System Controller, the default switch setting.

Caution   Disk cache programs may interfere with the LIF utilities access to the internal floppy disk drives and may cause disk read errors or program lockup. We recommend that disk cache programs not be used in conjunction with these utilities.

You can use the SMARTDrive disk caching software provided with Microsoft Windows 3.0.

Note   The utilities will work with both an HP and a National card installed simultaneously. For example, you can LIF copy files from a disk drive on the HP card to a disk drive on the National Instrument card.

Both BDAT and ASCII LIF files may be converted to DOS. (Binary LIF files are copied in raw mode.)

Note   When using these utilities, remember the following:
* The LIF utilities do not format internal disks (only external).
* It's not possible to initialize a LIF directory onto an unformatted disk.

Online Help

Online Help is available for each LIF utility command. For a description of a particular command and a list of its options, simply type the command name and /U. Then press return.

For example, to get help on the command LIF CHK, type:

LIF CHK /U [**Enter**]

LIF

Main program for LIF access. The first parameter specifies operation. Each operation has its own set of files/options. For more help on a particular command, see the rest of this chapter.

Syntax

LIF <operation> [files] [options]

Operations

SCAN   look for LIF disk drives (external or internal)

CHK or CHECK   check disk integrity (read all tracks/heads)

LS or DIR   list files on LIF disk

CP or COPY   copy file(s) between DOS and LIF disks

RM or DEL   remove a LIF file

INIT   initialize (format) a LIF disk

The specified <operation> has its own set of files/options. For more help on a particular command, type ''lif'' followed by the operation name (except for SCAN) and /U. For example, for help on the LIF CP command, type: LIF CP /U

LIF SCAN

Checks the PC for internal floppy drives and for HP-IB interface boards and then looks for disk drives (capable of reading LIF disks) connected to HP-IB.

This command displays the drive specifiers of all valid disk drives which may contain LIF files. Either a Hewlett-Packard HP-IB card or a National Instrument GPIB-PCII (or AT-GPIB) card (or compatible) will be recognized.

The drive specifier for the HP-IB disk drive consists of an address in the form SSAA[UU], where:

- SS is the select code.
  For the HP card the select code is from 1 to 16.
  For the National Instrument card, the select code is 1 plus the board number (for example, 1 for board ''GPIB0'' and 2 for board ''GPIB1'').
- AA is the HP-IB address of the disk drive (00 to 07).
- [UU] is the optional unit of the disk drive (00 to 15). The default unit is 0.

Syntax

> LIF SCAN [/U]

Options

/U   Show usage information for this command.

Example

Type LIF SCAN from your home directory, for the following information:

```
     Drive
    Specifier      Looking for Disk Drives
    =========      ======================

     A:         Internal Flexible Disk
     B:         Internal Flexible Disk

             Found HP 88500A HP-IB Card at Select Code 7
                 Found SS/80 Disk at Address 2, HP 9122
    702:             Unit 0, Flexible or Removable Disk
    70201:            Unit 1, Flexible or Removable Disk
```

Discussion

''702:'' identifies a disk drive at address 2 and unit 0 connected to an HP card at select code 7. If both an HP card and an National Instrument card are connected to the same select code, then only the HP card will be recognized. This is usually not a problem since the customary select code of the HP card is 7, and the customary board name for the National Instrument card is ''GPIB0'' (select code 1). The LIF programs are capable of working with both boards installed simultaneously (e.g., LIF copy from a disk drive on the HP card to a disk drive on the National Instrument card).

LIF SCAN indicates how to address a LIF disk in an external drive, not a DOS disk. If a DOS disk is in your external drive, you cannot access it as ''702:'' or ''70201:''. Instead, use the letter drive code that DOS has assigned (e.g., E: or F:).

MS-DOS requires that an external drive be powered up and connected at power-on to be recognized as an MS-DOS drive. The LIF program do not require external drives to be on at power-on to be accessible as a LIF drive.

LIF CHK

Verifies that the disk inserted in the specified drive is fully readable by the LIF program. This serves two functions: first, to verify that the disk has no bad sectors (to minimize the possibility of corrupted data), and second, to verify that possible spared tracks on the disk will not interfere with the disk's usage on the PC's internal disk drives.

Syntax

LIF CHK <d>: [/U] [/T] [/A] [/W]
or
LIF CHECK <d>: [/U] [/T] [/A] [/W]

Options

The default mode (no option) reads one sector on every track and displays the head and track number.

<d>: The drive specifier, e.g., 702: or 70201: for an external drive; A: or B: for an internal drive. All LIF files require a drive specifier.

/U Show usage information for this command.

/T Terse, do not show the Head and Track numbers as each track is read. This can decrease the verify time by up to 50%.

/A Read every sector on each track instead of just one sector on each track. This is useful if you believe that the disk has possible media problems (not useful for additional checking for spared tracks). This can increase the verify time by up to 50%.

/W Wait for keyboard prompt before and after performing operation. This is useful for a PC with only two disk drives; you can swap out the disk with the LIF.EXE program to insert a LIF or DOS disk.

Example

LIF CHK 702: /A

Discussion

HP spared tracks on an HP 9122 or HP 9127 disk may not be readable on the internal drive because the spared track may actually reside on a different side (head) of the disk and the PC BIOS may not be capable of finding it. This program verifies that each track can be read by the internal drive.

LIF LS

Lists the contents of the LIF directory contained on a designated drive. All files are listed in the order that they are contained on the disk.

Syntax

> LIF LS <d>: [/U] [/L] [/A] [/W]
> or
> LIF DIR <d>: [/U] [/L] [/A] [/W]

Options

The default option lists all normal (not PURGED) files.

<d>:   The drive specifier, e.g., 702: or 70201: for an external drive, A: or B: for an internal drive.

/U   Show usage information for this command.

/L   List in long format, giving volume name, volume size, directory start (in decimal), directory size, file type, file size, file start (in decimal), ''implementation'' field (in hex) and date created. (The implementation field indicates if a security code is present. The code is ignored by the LIF program.)

/A   List all entries including purged files (PURGED type). Purged files are not accessible by LIF CP.

/W   Wait for keyboard prompt before and after performing operation. This is useful for a PC with only two disk drives; you can swap out the disk with the LIF.EXE program to insert a LIF or DOS disk.

In the long format, the following special file types are named:

1	ASCII
0	PURGED (only for /A list option)
–2	BINARY
–767	DOS (DOS Directory)
–5791	BDAT

Example

List directory in long format:

> LIF LS 702: /L

Route directory listing to a file in your home directory:

> LIF LS 702: /A > FILELIST

Discussion

HP 3563A (or HP 3562A) files are all of type BDAT and the first two letters of the file name specifies the file type as follows:

A1-A5	Aseq1 through Aseq5
AM	Amath
CF	Cv Fit (S domain Curve Fit)
CS	Capture
DA	Data
DM	Demod (Delete Frequency)
FL	Fault (Fault Log)
SE	State
ST	Synth (S domain)
TS	Thrupt
TL	Test (Test Log)
ZF	Z CvFt (Z domain Curve Fit)
ZT	Z Snth (Z domain Synth)
PI	Pictur (Display graphics)

LIF CP

Copies a LIF file to a DOS file, a DOS file to a LIF file, or a LIF file to another LIF file. <sfile> is the source file name and <dfile> is the destination file name. The full path name must be specified for a LIF file. LIF ASCII and BDAT formats are automatically converted to DOS format. All other files are copied with no conversion. At least one of the files must be in a LIF directory (source or destination file). This may be one of the internal flexible disk drives (e.g., A: or B:) or it may be an external HP-IB disk drive (e.g., 700: or 70001:).

Syntax

> LIF CP <sfile> <dfile> [/U] [/O] [/B] [/A] [/R]
> [/T:<fileType>] [/I:<implement>] [/N] [/S:<lineSize>] [/W]
> or
> LIF COPY <sfile> <dfile> [/U] [/O] [/B] [/A] [/R]
> [/T:<fileType] [/I:<implement>] [/N] [/S:<lineSize>] [/W]

Options

<sfile>	Source path and file name (must include drive specifier if file is LIF). May contain wild card characters ''*'' and ''?''.
<dfile>	Destination path and file name (must include drive specifier if file is LIF). May contain wild card characters ''*'' and ''?''.
/U	Show usage information for this command.
/B	BDAT file copy (use only for DOS to LIF).
/A	ASCII file copy (use for DOS to LIF).
/R	Raw file copy (DOS to LIF or LIF to DOS). Do not convert file (copy as is).
/T:<fileType>	(LIF destination only) Set the file type for the destination file to the specified file type. This may be specified in decimal or hexadecimal (prefix with ''0x''). The file type specified will not affect any possible file conversions.
/I:<implement>	Set the implementation field for the destination LIF file (DOS to LIF or LIF to LIF). This may be specified in decimal or hexadecimal (prefix with ''0x'').
/N	Do not translate to valid LIF file name.
/S:<lineSize>	Maximum line size for DOS to LIF ASCII file transfer (default 256 characters).
/W	Wait for keyboard prompt before and after performing operation. This is useful for a PC with only two disk drives; you can swap out the disk with the LIF.EXE program to insert a LIF or DOS disk.

Example

LIF to DOS copy from an external drive to your current PC directory:

> LIF CP 702:DAFREQ DSK_FREQ

DOS to LIF copy from home directory to external drive. /N option is used to allow the LIF file name to begin with a number. If /N were not typed the file name would be X77PSD.

Note   Invalid LIF filenames may cause files to be inaccessible by another LIF system, e.g., Pascal workstations or HP BASIC.

> LIF CP 77PSD 702:77PSD /N

DOS to LIF copy from DOS external drive to LIF external drive:

> LIF CP 702:DAFREQ E:FREQ

DOS to LIF copy forcing long line length:

> LIF CP LONG.ASC 702:LONGLINES /S:1000

Discussion

Any specified LIF file name is automatically translated to contain a valid LIF file name as follows:

- Shifts lower to upper case.
- Allows ''_'', ''.'', and digits (0-9).
- Translates all other characters to the letter ''X''.
- If the first character is not a letter, then the file name will be preceded by the letter ''X''.
- Truncates file names to 10 characters.

Only two types of file conversions (i.e., not raw copy) are allowed between LIF and DOS files: ASCII and BDAT. LIF Binary files are not converted.

For a LIF to DOS file copy, LIF ASCII files are converted to DOS ASCII files and LIF BDAT files are converted to DOS Binary files. All other files are copied in raw mode (exactly as they appear in the LIF file). The automatic conversions for LIF ASCII and BDAT files may be turned off with the raw option (/R).

If a directory is specified as a destination, then a file with the same name as the source file will be created. If only a directory is specified as a source, then all files in the directory will be copied. For a DOS to LIF file copy, the file will be converted to either ASCII or BDAT automatically if no format conversion is specified (/A, /B, or /R).

If the first 80 characters of the DOS file contain only printable ASCII characters (or tab, carriage return, or line feed), then the file will be converted to LIF ASCII, otherwise it will be converted to LIF BDAT. The file may be forced to be converted to LIF ASCII with the ASCII option (/A). A file may be forced to be converted to BDAT format by specifying the BDAT option (/B) and a raw copy (no conversions) can be specified with the raw option (/R).

DOS file	™	LIF file
BINARY (no option)	becomes	BDAT
ASCII (no option)	becomes	ASCII
BINARY (with /A option)	becomes	ASCII
ASCII (with /B option)	becomes	BDAT

LIF file	™	DOS file
BDAT	becomes	BINARY
ASCII	becomes	ASCII

Caution

If the input file is ASCII for a DOS to LIF copy, the input lines will be truncated to the value set by /S option (default is 256 characters). If truncations occurs, the following message will appear:

Line "X" truncates from "m" to "n" characters.

where ''X'' is the line that was truncated, ''m'' is the original line length, and ''n'' is the truncated line length.

Use the /S option to increase the maximum input line length. Note that the /S option does not need to be specified for LIF to LIF or LIF to DOS file copies.

For a LIF to LIF copy, all files are copied as is (no conversion) with the destination file created with the same file type as the source file.

Multiple files may be copied if the source file specification is a directory or the wildcards ''*'' (match 0 or more of any character) or ''?'' (match 1 occurrence of any character) are used. In this case, the destination file specification should also be a directory or contain matching wildcards. Each file will be copied to this directory. If the destination is not a directory and doesn't contain wildcards, then all files will be copied to the same destination file (not appended). This is only useful to copy files to your screen or to a printer, as LIF CP will not write to an existing file (except CON or PRN devices).

Example 1

Copy all the DOS files in the directory \DOSdir to the LIF disk at 702:

LIF CP a:\DOSdir 702:

Example 2

Copy all the DOS files with the suffix ''.doc'' to LIF disk at 702:

LIF CP *.doc 702:

Example 3

Copy all HP 3562A data files (prefix ''DA'') from a LIF disk at 702: to the current DOS directory, changing the ''DA'' prefix to a ''.63'' suffix.

LIF CP 702:DA* *.63

LIF RM

Removes (deletes) the specified file from the LIF directory.

Syntax

LIF RM <file> [/U] [/N] [/F] [/W]
or
LIF DEL <file> [/U] [/N] [/F] [/W]

Options

<file>   File name. Must include drive specifier (702:, A:, etc.).

/U   Show usage information for this command.

/N   Do not translate to valid LIF file name.

/F   Force remove (i.e., external DOS directory).

/W   Wait for keyboard prompt before and after performing operation. This is useful for a PC with only two disk drives; you can swap out the disk with the LIF.EXE program to insert a LIF or DOS disk.

Example

Look for a file called 00DATA. If it doesn't exist, it won't remove any files.

LIF RM 702:00data

Remove a file with the name ''00data''.

LIF RM 702:00data /N

Discussion

When you type in the name of a file (except where /N is specified), this utility automatically converts a valid LIF file name as follows:

- Shifts lower to upper case.
- Allows ''_'', ''.'', and digits (0-9).
- Translates all other characters to the letter ''X''.
- If the first character is not a letter, then the file name will be preceded by the letter ''X''.
- Truncates file names to 10 characters.

Caution   Be careful with this command; a DOS directory on an external HP-IB disk drive (for example, an HP 9133D) is actually a LIF directory with one entry (for example, VOLUME000) and can be removed with this utility only by specifying the /F option.

LIF INIT

The command LIF INIT optionally formats and initializes a new directory on a LIF disk.

Syntax

LIF INIT <d>:[name] [/U] [/F] [/O:<opt>] [/I:<intrlv>] [/W]

Options

<d>:   Drive specifier (702:, A:).

[name]   Optional new LIF directory name (up to 6 characters).

/U   Show usage information for this command.

/F   Format disk before initializing directory. This operation may take a few minutes to complete. This operation may not be necessary if the disk is already formatted. Disks can only be formatted on external HP-IB disk drives.

/O:<opt>   Format option. This integer number specifies the disk sector size and other disk drive dependent information. Refer to your disk drive owner's manual for possible values and their specific meaning. (If you have an HP 9122 or HP 9127, see the formatting options on the following pages.) If not specified, then a format option of 0 is used. The format option is only used if format disk (/F) is specified.

/I:<intrlv>   Sector interleave factor. A value of 1 is the default. The interleave factor is only used if format disk (/F) is specified.

/W   Wait for keyboard prompt before and after performing operation. This is useful for a PC with only two disk drives; you can swap out the disk with the LIF.EXE program to insert a LIF or DOS disk.

Caution   Do not eject disk while formatting.

Example

Format a disk in an external drive:

> LIF INIT 702:MIKE /F

Format a single-sided disk on an HP 9122D:

> LIF INIT 702:HALL /F /O:4 /I:2

Discussion

The [name] specified (in this case ''HALL'') is the new LIF directory's volume name and is automatically translated to contain a valid LIF volume name as follows:

- Shifts lower to upper case.
- Allows ''_'', ''.'', and digits (0-9).
- Translates all other characters to the letter ''X''.
- If the first character is not a letter, then the volume name will be preceded by the letter ''X''.
- LIF INIT truncates file names to 6 characters. If no name is specified, then the LIF volume name will be blank.

Disks can only be formatted on external HP-IB disk drives (i.e., not on internal flexible disk drives).

Note   Any LIF disk used with the HP 3563A/62A must have a non-empty LIF Volume name. This means that if you are formatting a disk on the PC with LIF INIT, always specify the ''Optional new LIF directory name.''

HP 9122D Formatting Options

If you have an HP 9122D and wish to use it to format and initialize disks with the LIF program, make sure you use the following format options.

Format Option	Bytes/ Sector	Double-or Single-sided Formatting	Kbytes of Storage
0,1	256	Double-sided	630K
2	512	Double-sided	710K
3	1024	Double-sided	788K
4	256	Single-sided	270K

HP 9127A Formatting Options

If you have an HP 9127A and wish to use it to format and initialize disks with the LIF program, make sure you use the following format options.

Format Option	Bytes/ Sector	Double-or Single-sided Formatting	Kbytes of Storage
0,1	256	Double-sided	270K
3	1024	Double-sided	378K

Do not use format options 2, 4, 5, or 6.
This will create a DOS format disk which is incompatible with LIF.

Caution   The HP 3562A and HP 3563A only support a sector size of 256 Bytes.

LIFDIAG

LIFDIAG contains diagnostic functions useful for recovering data from damaged/corrupted disks and to perform fast full LIF to LIF disk backups.

The menu line for LIFDIAG contains a menu name followed by a set of command words on the top line on the display. Each command has one letter in red on a color display (bright white on monochrome display) indicating the hot key for that command. Pressing the hot key causes the command to be executed. The currently selected command is in inverse video on a color display (in bright white on monochrome display). As an alternative to using the hot key to execute a command, the currently selected command may be moved by pressing the left or right arrow keys and executed by pressing the [**Enter**] key.

LIFDIAG Menu:

LIFDiag:	**E**dit	**B**ackup	**H**elp	**Q**uit
Hot Keys:	E	B	H	Q

Edit
: Edit logical sectors on a LIF disk, including read sectors (with display in hex/ASCII), write sectors (even to other disk addresses), modify sector contents (in hex, decimal, or ASCII), and find data pattern on disk (specify in hex, decimal, or ASCII). Any number can be specified in hex by prefacing the number with ''0x''. Pressing this key will prompt you to enter the disk path of the LIF disk to edit. If an error results in specifying the disk path (e.g. no disk), then the edit menu will not allow the disk to be edited. Press ''Quit'' and try ''Edit'' again. The size & characteristics of the disk are displayed if no errors are detected.

For a full description of edit functions, see ''EDIT Menu.''

Backup
: Full disk backup between specified source and destination disk (can be the same disk). Either the entire disk may be copied, or just the contiguous file area contained by the LIF files. For damaged disks, there is an option to skip bad sectors that are read/written (they are zeroed out on the destination disk). Any errors are displayed in a special ERRORS window. For a full description of Backup functions, see ''BACKUP Menu.''

Help
: Display command summary on the display.

Quit
: Quit LIFDIAG. Pressing this key will ask for confirmation. Press ''Yes'' to exit the program.

EDIT Menu:

Edit:	Read	Next	Prev	Find	Modify	Write	Log	Help	Quit
Hot Keys:	R	N	P	F	M	W	O	H	Q

Read   Read/display the logical sector specified. Logical sectors are always 256 bytes even though physical sectors may be 256, 512 or 1024 bytes long. For example, if the disk has 1024 byte physical sectors, then physical sector 0 is contained in logical sectors 0 through 3. The sector is displayed in hexadecimal (on left side of display) and in printable ASCII (on right side of display).

Next   Read/display the next logical sector (current sector + 1).

Prev   Read/display the previous logical sector (current sector - 1).

Find   Find the specified data pattern between the specified start and stop sectors on the disk. The data pattern may be specified as a string of up to 70 ASCII characters (case sensitive) or up to 35 comma or space separated numbers representing individual bytes. Each number may be specified in decimal or hexadecimal (prefaced with ''0x''). This operation can be aborted by pressing the ''Esc'' key.

Modify   Modify the currently displayed sector. The offset from the start of the sector is specified as are the new values to place in the sector. The values may be ASCII characters or comma/space separated numbers (bytes).

Note

The sector is not modified on the disk until a Write command is performed.

Write   Write the currently displayed sector to the disk, either at the same sector or a different sector on the disk. The logical sector address may be specified in decimal or hexadecimal (prefaced with ''0x''). The default sector number is the last sector that was read.

Log   Specifies a log file where the displayed hex/ASCII sector contents will be appended
                        to. For example, ''prn'' specifies the printer.

Help   Display command summary on the display.

Quit   Quit editing this disk.

Caution   Always select ''Quit'' and ''Edit'' again before editing a new disk. The
                        size & characteristics of the disk are only determined when the disk is initially edited.

BACKUP Menu:

Backup:	**G**o	**S**ource	**D**est	**L**ength	**E**rrors	**Lo**g	**H**elp	**Q**uit
Hot Keys:	G	S	D	L	E	O	H	Q

Go
Start the backup. If the source and destination disks are the same, then you will be prompted to swap source/dest disks. During the backup, the PROGRESS window shows how much of the backup operation has been completed. A backup may be aborted at any time by pressing the ''ESC'' key.

Source
Specify the external (e.g. 102:) or internal (e.g. b:) source LIF disk. The source disk should be write protected to prevent accidental erasure of data in the event that the disk is placed in the wrong drive.

Dest
Specify the external (e.g. 102:) or internal (e.g. b:) destination LIF disk. The destination disk must be writeable and already formatted. The destination disk must be large enough to contain the contents of the source disk. A warning will be displayed if the destination is too small for full disk copy.

Length
Specify whether the entire source disk (Full disk) will be copied or only the area of the disk containing up to the last LIF file. Full disk should be specified if the LIF directory has been damaged.

Errors
Specify whether a read/write error will cause the backup to Stop or whether the error is Ignored (the sector is cleared on the destination disk). Any errors are displayed in the ERRORS window.

Log
Specifies a log file where the same errors that are shown in the ERRORS window will be appended to. For example, ''prn'' specifies the printer. Note: the log file must be specified before starting the backup.

Help
Display command summary on the display.

Quit
Quit backup function.

Error Messages

Cannot format
internal disk

The internal disks on the PC cannot be formatted by LIF INIT. Use an
HP-IB disk drive to format your disks, if available, or format your disks
on another machine.

Disk access
(Check media)

Access errors can occur on media that is possibly corrupted.
Immediately, try to copy all files off the disk, or just dispose of the disk
and use a backup disk. The disk may be salvageable (not the data) if it is
re-formatted.

Disk fault (Check
disk drive)

Fault errors indicate disk drive hardware failures. Try running your disk
drive diagnostics to trace the problem, or just cycle power on the disk
drive (a hard reset).

Disk has no media

No disk is in the disk drive being accessed. It is possible that the wrong
disk address is being specified by the user. Double check the drive
specifier you are using. A common mistake when using the HP 9122
disk drive is incorrectly specifying the disk unit (unit 0 is left drive, unit
1 is right drive).

Disk is not SS/80

Older HP-IB disk drives (Amigo Command Set) can not be used with the
LIF program. These disks include the HP 9121, HP 9133 (V, X, or XV),
HP 82901, or HP 9895. Try using a newer disk drive such as HP 9122,
HP 9133 (D, H, or L), or HP 9127A.

Disk is write
protected

The target disk for a LIF CP, LIF RM, or LIF INIT is write protected.
Check the disk to make sure you really want to modify it, and remove
the write protect mechanism on the disk.

Disk lost power

During a LIF utility operation, the disk drive either lost power or the disk
was ejected and replaced. This could corrupt data on your disk if a disk
was swapped during a LIF CP operation. Do not eject disks during any
LIF utility operation.

Disk media is unformatted

The disk may actually be a DOS disk. Try using the DOS command
''dir'' to check whether this disk is in DOS format. If not, then format
the disk (LIF INIT /F) before using it.

Disk reject (Check connections)

Reject errors generally indicate a failure in trying to communicate with
an external disk drive. Check the HP-IB cables to make sure they are
secure. Also, try disconnecting any other HP-IB devices which may be
interfering (e.g., same HP-IB address) with communication to the disk
drive. It is also possible for this error to occur if the LIF directory is
corrupt.

Disk timeout

The external disk drive did not respond in the time allowed. This can occur if no disk drive is at the specified address, the disk is not connected properly, the disk drive is not turned on, or another device on the HP-IB is interfering with communication.

DOS directory

The external disk drive contains a DOS directory (it actually is a valid LIF directory which contains one file, which is the DOS directory). Be very careful with this directory. If you delete the directory file (via LIF RM) or reinitialize the disk (via LIF INIT), then your DOS directory is lost. You can actually remove this directory by specifying the /F option to LIF RM.

File already exists

The LIF CP utility generates this error if the target file already exists on the LIF disk. Use ''LIF RM'' to remove the existing file from a LIF disk or ''del'' to remove the existing file from a DOS disk.

File not found

The source file was not found by LIF RM or LIF CP. Check the spelling of the file name. Remember that LIF file names are only 10 characters long. File names are automatically translated to valid LIF names. If the file name desired is not a valid LIF name, then use the /N option to suppress the name translations.

LIF file system is full

Either the directory area or the data area of the disk is full. Either remove some files from the target disk, or use another disk which has room.

LIF internal error

This error indicates a possible problem with the PC's memory.

Neither file is in a LIF file system

This error is generated by LIF CP if neither the source nor the destination file is in a LIF file system. Check your path names. Remember that only internal floppies and external disk drives may be LIF file systems.

No disk drive

No disk drive was found at the specified unit, but there is a disk drive at the specified address. The disk unit specifier was probably specified incorrectly.

Not a valid drive specifier

The drive specifier was incorrect. Run LIF SCAN to find all valid LIF disk drive specifiers.

Not a valid HP-IB card address

There is no HP HP-IB or National Instrument GPIB-PCII card at the specified address. Run LIF SCAN to verify the HP-IB card address.

Not a valid LIF directory

The disk does not contain a valid LIF directory, but it does contain a valid LIF volume header. Check your disk, it is probably corrupt.

Not a valid LIF file system	The disk does not contain a valid LIF volume header. Check your disk, it is either a DOS disk, or the disk is corrupt.
Not a valid format option	The format option specified on the LIF INIT command is not valid for this disk drive. For example, only the format options 0 through 4 are valid on an HP 9122 disk drive. Refer to your disk drive operator's manual for allowable format options. For HP 9122 or HP 9127 format options, see command LIF INIT.
Not a valid interleave	The interleave value specified on the LIF INIT command is not valid for this disk drive. The allowable values for interleave are 0 through X-1 where X is the number of usable sectors per track. For example, an HP 9122 with 256 byte sectors has 16 sectors per track, so an interleave of 0 through 15 is valid.
Out Of Memory	Not enough memory to read the disk directory or to contain the file system control structures. Free up some memory.
Sector size not supported	Sector size of 512 bytes for a 5 1/4" drive is not usable because it can only be formatted (on a HP 9127A) as DOS format (not LIF). If it is a 3.5" disk, the disk was not formatted with an HP-IB disk drive (HP 9121 or HP 9122 compatible) or the disk is corrupt.
Unknown Option 'X'	This error may be generated by all the commands. Check your options. Typing the command with /U will list the valid options. Options are not case sensitive or order dependant.

Sharing Data Between Analyzers

Introduction

The Standard Data Format Utilities are programs that translate data files from the HP 35665A, HP 3560A, HP 3563A, HP 3562A, HP 3566A, HP 3567A, HP 3588A, or HP 3589A analyzers to Standard Data Format or formats compatible with ASCII, Data Set 58, PC-MATLAB, and MATRIXx.

Conversions from Standard Data Format to STARModal and STARAcoustics formats are available from Structural Measurement Systems.

With the Standard Data Format utilities you can share data between the supported HP DSA analyzers using these general steps:

1. Make a measurement with one of the supported HP DSA analyzers.

2. Save the measurement data.

3. Convert the data using the appropriate conversion utility.

4. Recall the data into another analyzer.

5. Do further analysis on the data, or compare it to other data.

For example, you can make a frequency response measurement with an HP 3560A, transfer the data to your PC, convert it to SDF format, then recall it into an HP 35665A for curve fitting (see figure 1-2).

Or you can compare old data from an HP 3562A to new data from an HP 35665A (see figure 1-3).

PC-MATLAB (a trademark of The MathWorks, Inc.) and MATRIXx (a product of Integrated Systems Inc.) are PC-based software packages for scientific and engineering numeric computation. HP SDF data files may be ported to either of these packages using the Conversion Utilities. See appendix A for examples of how to load SDF data into these application packages.

The utilities are as follows:

60TOSDF	Convert an HP 3560A data file to an SDF data file.
63TCSDF	Convert the data portion of a binary capture/throughput file from the HP 3563A (or HP 3562A) to SDF time capture format.
63TOSDF	Convert an HP 3563A (or HP 3562A) data file to an SDF data file.
660TOSDF	Convert an HP 35660A data file to an SDF data file.
69TOSDF	Convert an HP 3569A data file to an SDF data file.
88TOSDF	Convert an HP 3588A data file to an SDF data file.
89TOSDF	Convert an HP 3589A data file to an SDF data file.
ASCTOSDF	Convert an ASCII data file to an SDF data file.
BINTOX32	Convert data from the HP 3569A source binary file and put the results in the destination Intel Hex-32 data file.
DFDATA63	Convert the data portion of a binary capture/throughput file from the HP 3563A (or HP 3562A) to either an ASCII file, a PC-MATLAB MAT-file, or a MATRIXx file.
DFHDR63	Convert the header portion of a binary capture/throughput file from the HP 3563A (or HP 3562A) to an ASCII file.
DOWNLOAD	Transfer data from an HP 3560A over the personal computer's RS-232 port and place it in an output file.
HEADER63	Convert the data header portion of a binary data file from the HP 3563A (or HP 3562A) to an ASCII form.
HPIB63	Transfer data from an HP 3562A (or HP 3563A) over the personal computer's HP-IB and place it in an output file.
REPEAT	Repeatedly calls the specified program.
SDFEDIT	Edit any selected SDF header fields of an SDF file.
SDFPRINT	Print, in ASCII form, all or selected portions of an SDF file.
SDFTEXT	Edit/print the comment text of an SDF file.
SDFTO58	Convert data from the SDF format to Data Set 58.

SDFTO63   Convert a single measurement result from an SDF data file to an HP 3563A (or HP 3562A) data file.

SDFTOASC   Convert an SDF data file to an ASCII data file.

SDFTOBIN   Convert an SDF data file to a raw binary data file.

SDFTOML   Convert data from the SDF format to the format used in PC-MATLAB.

SDFTOMX   Convert data from the SDF format to the format used in MATRIXx.

SDFTOSDF   Extract a subset of results from an SDF file and create another SDF file. This is not necessary for all analyzers. Because of differences in the way analyzers store data, you may need to run this program (see chapter 2). Sharing data between analyzers may require you to convert the number of lines of resolution to a different number of lines. You may also need to change the arbitrary spacing (from a swept sine measurement) to log or linear spacing (for curve fit or math operations).

SDFYDATA   Edit/print any section of Y data of an SDF file.

SETUP63   Convert a binary setup state file from the HP 3563A (or HP 3562A) to an ASCII form.

SETUP88   Convert a binary setup state file from the HP 3588A to an ASCII form.

SETUP89   Convert a binary setup state file from the HP 3589A to an ASCII form.

SOFTCOPY   Transfers data from the HP-IB (IEEE-488) to a file.

SYNTH63   Convert a binary synthesis file from the HP 3563A (or HP 3562A) to an ASCII file, a PC-MATLAB MAT-file, or a MATRIXx file.

VIEWDATA   Displays data graphically on your personal computer. See the chapter in this book titled ''Viewdata'' for more information.

X32TOBIN   Convert an Intel Hex-32 data file to a binary data file. (Intel Hex-32 is the internal data format for the HP 3560A.)

Note   These utilities work only on MS-DOS files. Prior to using them, copy your HP 3563A/3562A files from LIF to DOS using the LIF utilities (LIF CP) described in the first part of this documentation.

Hardware and Software Requirements

Use of these utilities requires an IBM PC compatible with the MS-DOS 2.1 operating system (or greater).

HP 3563A/3562A File Format

When the HP 3563A/3562A saves measurement data or instrument states to an HP-IB disk drive, the files are saved in a non-standard binary format used by the instrument. Typically, you need to transfer three pieces of information: your setup state, file header information, and measurement data. A setup state will be in a single file, while the header information will be attached to a data file. Using these utilities, you can separate header and data information into two files.

When you look at these files using the LIF utility command LIF LS /L, you'll see that the analyzers files are all type BDAT. The first two letters in the file name indicate the file type:

File type:	File name starts with:
Data	DA
Setup State	SE
Synthesis	ST
Curve Fit (Z domain)	CV ZF
Capture	CS
Throughput	TS

Online Help

Online Help is available for each conversion utility command. For a description of a particular command and a list of its options, simply type the command name and /U.

For example, to get help on the command 63TOSDF, type:

63TOSDF /U [**Enter**]

60TOSDF

Convert data from the HP 3560A source data file and put results in the destination SDF file. The destination SDF filename is chosen based on the register value that was used in the HP 3560A and is of the form <register number>.DAT (for example 1.DAT).

Syntax   60TOSDF <sfile> [/U] [/O] [/H] [/A] [/P:<c,p,d>]

Options

<sfile>   Input HP 3560A data file name.

/U   Show help (usage information) for this program.

/O   Overwrite <dfile> if it exists.

/H   Print headers to the screen.

/A   Print headers and data to the screen.

/P:<c,p,d>   Set point and direction for a channel (/P:), <c>is the channel number (0 or 1), <p> is the point number, and <d> is the direction as follows:

0	None	5	Theta
1	X	6	Phi
2	Y	7	TX
3	Z	8	TY
4	Radial	9	TZ

Example

Transfer an HP 3560A data file (which is in Intel Hex-32 format) to a file (FREQ.X32) on a personal computer using the RS-232 port. Convert the Intel Hex-32 file to a binary data file (FREQ.BIN) and then convert the binary file to an SDF data file. The data file name is the same as the register number on the HP 3560A (for example, 1.DAT).

```
DOWNLOAD FREQ.X32 /B:19200 /P:2
X32TOBIN FREQ.X32 FREQ.BIN
60TOSDF FREQ.BIN
```

See also the batch file ''60_SDF.BAT'' in appendix C.

63TCSDF

Convert the data portion of a binary capture/throughput file from the HP 3563A (or HP 3562A) to SDF time capture format (16-bit integer data). The binary capture/throughput file contains digital filter data. The HP 3563A/62A capture files on a LIF file system start with the letters ''CS''. Throughput files start with ''TS''.

Syntax   63TCSDF <sfile> <dfile> [/U] [/O]

Options

<sfile>   Input HP 3563A/62A capture or throughput file name.

<dfile>   Output SDF file name.

/U   Show help (usage information) for this program.

/O   Overwrite <dfile> if it exists.

Example

Copy a capture file (CSCAPTUR) from the HP 3563A LIF formatted disk drive (drive B:) to a file (CAPTUR.63) on the current hard drive and convert it to SDF format (CAPTUR.TIM):

                    LIF CP B:CSCAPTUR CAPTUR.63
                    63TCSDF CAPTUR.63 CAPTUR.TIM

63TOSDF

Convert an HP 3563A (or HP 3562A) data file <sfile> to an SDF data file and outputs to <dfile>.
HP 3563A/62A data files on a LIF file system start with the letters ''DA.''

Syntax   63TOSDF <sfile> <dfile> [/U] [/O]

Options

<sfile>   Input HP 3563A/62A data file name.

<dfile>   Output SDF file name.

/U   Show help (usage information) for this program.

/O   Overwrite <dfile> if it exists.

Example

Copy a data file (DAFREQ) from the HP 3563A LIF formatted disk (drive B:) to a file (FREQ.63) on the current hard drive and convert it to SDF format (FREQ.DAT):

```
LIF CP B:DAFREQ FREQ.63
63TOSDF FREQ.63 FREQ.DAT
```

Discussion

A baseband time record contains 2048 real points, a zoom time record contains 1024 complex points, and a spectrum contains either 801 real or complex points or 1024 real or complex points.

See also the batch file ''63_SDF.BAT'' in appendix C.

660TOSDF

Extract data from an HP 35660A data file <sfile> and put results in the SDF data file <dfile>.

Syntax   660TOSDF <sfile> <dfile> [/U] [/O] [/H] [/A]

Options

<sfile>   Input HP 35660A data file name.

<dfile>   Output SDF file name.

/U   Show help (usage information) for this program.

/O   Overwrite <dfile> if it exists.

/H   Print headers to the screen.

/A   Print headers and data to the screen.

Example

Copy a data file (TRACE1) from the HP 35660A LIF formatted disk (drive B:) to a file (FREQ.660) on the current hard drive, then convert the file to SDF format (FREQ.DAT):

        LIF CP B:TRACE1 FREQ.660
         660TOSDF FREQ.660 FREQ.DAT

69TOSDF

Convert data from the HP 3569A source data file and put results in the destination SDF file. If [dfile] is not specified, the destination SDF filename is chosen based on the filename that was used in the HP 3569A and is of the form <filename>.DAT (for example, 1.DAT).

If the input X32 file contains more than 1 HP 3569A trace file (SAVE/RECALL, XFER ALL TRACES), then each trace will be put in a different file with the same base name as used on the HP 3569A. If [dfile] is specified, only the first trace in the file is used.

Syntax   69TOSDF <sfile> [dfile] [/U] [/O] [/H] [/A] [/P:c,p,d][/M:<value>][/D:<value>]

Options

<sfile>   Input HP 3569A data file name.

[dfile]   Optional output SDF data file name.

/U   Show help (usage information) for this program.

/O   Overwrite [dfile] if it exists.

/H   Print headers to the screen.

/A   Print headers and data to the screen.

/P:c,p,d   Set point and direction for a channel (/P:).   <c>is the channel number (0 or 1), <p> is the point number, and <d> is the direction as follows:

0	None	5	Theta
1	X	6	Phi
2	Y	7	TX
3	Z	8	TY
4	Radial	9	TZ

/M:<value>   Set microphone separation for FFT Intensity data in mm.

/D:<value>   Set density for FFT Intensity data in Kg/m^3

Example

Transfer an HP 3569A data file (which is in Intel Hex-32 format) to a file (FREQ.X32) on a personal computer using the RS-232 port. Convert the Intel Hex-32 file to a binary data file (FREQ.BIN) and then convert the binary file to an SDF data file (FREQ.DAT).

        DOWNLOAD /B:19200 /P:2 FREQ.X32
        X32TOBIN FREQ.X32 FREQ.BIN
        69TOSDF FREQ.BIN FREQ.DAT

See also the batch file ''69_SDF.BAT'' in appendix C.

88TOSDF

Converts an HP 3588A data file <sfile> to an SDF data file and outputs to <dfile>. An HP 3588A setup state file <hfile> can be optionally specified which will allow more complete parameter conversion into the SDF data file.

Syntax   88TOSDF <sfile> <dfile> [/O] [/U] [/F <hfile>]

Options

<sfile>   Input HP 3588A data file name.

<dfile>   Output SDF file name.

/O   Overwrite <dfile> if it already exists.

/U   Show help (usage information) for this program.

/F   Read HP 3588A setup state file for additional information.

<hfile>   HP 3588A setup state path and file name.

Example

Copy a measurement data and setup state file from the HP 3588A LIF formatted disk and convert it to SDF format:

```
LIF CP B:TRACE1 TRACE1.88
LIF CP B:STATE1 STATE1.88
88TOSDF TRACE1.88 TRACE1.DAT /F STATE1.88
```

Discussion

When using the /F option, the setup state file <hfile> must contain the analyzer state information that represents the setup whenever the measurement data <sfile> was saved to disk.

If the /F option is not used in the conversion or the setup state file <hfile> does not reflect the setup whenever the measurement data <sfile> was saved, the following fields may not be set properly in the SDF file <dfile>.

Measurement Header	Channel Header
videoBandWidth	inputImpedance
averageType	range
averageNum	window.windowType
detection	window.windowBandWidth (Resolution BW)
	window.windowCorrMode
	window.narrowBandCorr
	window.wideBandCorr

When the HP 3588A saves measurement data or instrument states to the floppy drive, the files are saved in a non-standard binary format used by the instrument. Typically, you need to transfer three pieces of information:

- Setup state
- File header information
- Measurement data

A setup state is in a single file. The header information is part of a measurement data file. The 88TOSDF utility lets you combine the setup state and measurement data file into an SDF data file.

Note   The LIF utility command LIF LS /L will show all HP 3588A files as type BDAT. The analyzer saves setup state files with the [SAVE STATE] softkey. It saves measurement data files with the [SAVE TRACE] softkey.

89TOSDF

Converts an HP 3589A data file <sfile> to an SDF data file and outputs to <dfile>. An HP 3589A setup state file <hfile> can be optionally specified which will allow more complete parameter conversion into the SDF data file.

Syntax   89TOSDF <sfile> <dfile> [/O] [/U] [/F <hfile>]

Options

<sfile>   Input HP 3589A measurement data path and file name.

<dfile>   Output SDF path and file name.

/O   Overwrite <dfile> if it already exists.

/U   Show help (usage information) for this program.

/F   Read HP 3589A setup state file for additional information.

<hfile>   HP 3589A setup state path and file name.

Example

Copy a measurement data and setup state file from the HP 3589A LIF formatted disk and convert it to SDF format:

    LIF CP B:TRACE1 TRACE1.89
    LIF CP B:STATE1 STATE1.89
    89TOSDF TRACE1.89 TRACE1.DAT /F STATE1.89

Discussion

When using the /F option, the setup state file <hfile> must contain the analyzer state information that represents the setup whenever the measurement data <sfile> was saved to disk.

If the /F option is not used in the conversion or the setup state file <hfile> does not reflect the setup whenever the measurement data <sfile> was saved, the following fields may not be set properly in the SDF file <dfile>.

Measurement Header	Channel Header
videoBandWidth	inputImpedance
averageType	range
averageNum	window.windowType
detection	window.windowBandWidth (Resolution BW)
	window.windowCorrMode
	window.narrowBandCorr
	window.wideBandCorr
	gateBegin
	gateEnd
	userDelay

When the HP 3589A saves measurement data or instrument states to the floppy drive, the files are saved in a non-standard binary format used by the instrument. Typically, you need to transfer three pieces of information:

- Setup state
- File header information
- Measurement data

A setup state is in a single file. The header information is part of a measurement data file. The 89TOSDF utility lets you combine the setup state and measurement data file into an SDF data file.

Note   The LIF utility command LIF LS /L will show all HP 3589A files as type BDAT. The analyzer saves setup state files with the [SAVE STATE] softkey. It saves measurement data files with the [SAVE TRACE] softkey.

ASCTOSDF

Convert the source file to the destination file and place the result in the destination file. The following options are only valid when used with the /T (template SDF file) option; /D, /R, /C, /S. The options described after the /T option are only used if a template SDF file is not specified.

Syntax   ASCTOSDF <sfile> <dfile> [/U] [/O] [/D:<data>] [/R:<row>] [/C:<col>[[/S:<scan>]
[/T:<sdffile>] [/H:<type>] [/B:<blksize>] [/L:<lines>] [/Z:<ctrFrq>] [/X:<s,d>[,L]]

Options

<sfile>   Input ASCII file.

<dfile>   Output SDF file.

/U   Show help (usage information) for this program.

/O   Overwrite <dfile> if it already exists.

/D:<data>   Select data for the specified SDF_DATA_HDR. The default is 0.

/R:<row>   Select data for the specified row. The default is 0.

/C:<col>   Select data for the specified column. The default is 0.

/S:<scan>   Select data for the specified scan. The default is 0.

/T:<sdffile>   Template SDF file. If you already have a file (for example, a frequency response file) set up, use this option to copy that file and replace the data with the input ASCII file.

/H:<type>   Create default headers, where type is:
                         Time – time data (the default).
                         LSpec – linear spectrum
                         PSpec – power spectrum
                         FRF – frequency response

/B:<blksize>   Block size. Use this when the header type is Time. The default is the number of points in the file.

/L:<lines>   Frequency lines. Use this when the header type is not Time. The default is the number of lines in the file.

/Z:<ctrFrq>   Turn on zoom and set Center Frequency (use when header type is TIME)

/X:<s,d>[,L]   Generate X data. Start and Delta values are required. The optional [L] signifies log. Otherwise linear is used.

Example

Convert an ASCII data file to an SDF file using the linear spectrum header and a 2048 block size:

ASCTOSDF TEST.TXT TEXT.DAT /H:LSPEC /L:800

BINTOX32

Convert data from the HP 3569A source binary file and put the results in the destination Intel Hex-32 data file.

Syntax   BINTOX32 <sfile> <dfile> [/U] [/O]

Options

<sfile>	Input binary file name.
<dfile>	Output Hex-32 file name.
/U	Show help (usage information) for this program.
/O	Overwrite <dfile> if it exists.

Example

Convert an HP 3569A binary data file (FREQ.BIN) to an Intel Hex-32 file (FREQ.X32):

        BINTOX32 FREQ.BIN FREQ.X32

DFDATA63

Converts the data portion of a binary capture/throughput file (containing digital filter time data) from the HP 3563A (or HP 3562A) to a PC-MATLAB MAT-file, a MATRIXx file, or an ASCII file. Since capture/throughput data can be very long, the /B and /P options can be used to select only a portion of the data in a file. The /C option only needs to be specified for a 2 channel throughput file to select channel 2 as the source of data. HP 3563A/62A capture files on a LIF file system start with the letters ''CS''; throughput files start with the letters ''TS.''

Syntax   DFDATA63 <sfile> <dfile> [/U] [/O] [/L] [/M] [/R] [/X] [/B:<ofst>] [/P:<len>] [/C:<chan>]

Options

<sfile>	Input HP 3563A/62A capture or throughput file name.
<dfile>	Output file name.
/U	Show help (usage information) for this program.
/O	Overwrites <dfile> if it exists.
/L	Create MATLAB file (a binary file).
/M	Create MATRIXx file (an ASCII file).
/R	Output MATLAB or MATRIXx file matrices in a row instead of a column.
/X	Output X-axis data.
/B:<ofst>	Offset from beginning of data (in points) to convert (default 0).
/P:<len>	Length of data (in points) to convert (default 20480 points); MATLAB and MATRIXx files can contain up to 32767 points.
/C:<chan>	Channel of data to convert (default channel 1 if there are 2 channels of data).

If neither /L nor /M option is specified, then an ASCII file is created.

Example 1

Convert up to 4096 points of your file (GLITCH) to a PC-MATLAB file (GLITCH.MAT); includes X-axis data points. Note the PC-MATLAB searches for a file with the suffix ''.MAT'' if no suffix is specified. See appendix for more information on loading a file into PC-MATLAB.

DFDATA63 GLITCH C:GLITCH.MAT /L /X /P:4096

Example 2

Output channel 2 data (THRUPUT) to an ASCII file (THRUPT.ASC) after skipping the first 20480 points of the data file:

DFDATA63 THRUPUT THRUPT.ASC /B:20480 /C:2

Discussion

Each capture/throughput record contains 2048 real points (baseband) or 1024 complex points (zoom). A capture file contains up to 10 records of time data, whereas a throughput file contains up to 32767 records. Since capture/throughput data can be very long, the /B and /P options can be used to select only a portion of the data in a file. The /C option only needs to be specified for a 2 channel throughput file to select channel 2 as the source of data.

An ASCII file contains one line for each data point (up to 3 fields separated by commas). If the /X option is specified, the first field is the X-axis data value. The next field is the real part of the Y data point. If the data is complex, then the last field is the imaginary part of the Y data point.

Both the PC-MATLAB and MATRIXx files contain the following matrix variables:

Trace X	Vector of X values (only if /X is specified).
Start X	Starting X value (only if /X is not specified).
IncrX	Linear X increment between points (only if /X is not specified).
LogX	Log X increment between points (only if /X is not specified). This value is always 1.0.
TraceData	Vector of Y values (may be complex).

If the /X option is not specified, the StartX and IncrX variables can be used to calculate the X data as follows:

$$X = (Xlast + IncrX)$$

where Xlast is the previous point's X and the first X is StartX.

DFHDR63

Converts the header portion of a binary capture/throughput file <sfile> (containing digital filter data) from the HP 3563A (or HP 3562A) to an ASCII form <dfile>. Refer to the HP 3563A or HP 3562A *Programming Manual* for descriptions of the individual fields in the capture/throughput file.
HP 3563A/62A capture files on a LIF file system start with the letters ''CS''; throughput files start with the letters ''TS.''

Syntax   DFHDR63 <sfile> [dfile] [/U] [/O] [/L:<flag>] [/E:<flag>]

Options

<sfile>   Input HP 3563A/62A capture or throughput file name.

[dfile]   Output file name.

/U   Show help (usage information) for this program.

/O   Overwrites [dfile] if it exists.

/L:<flag>   If <flag> is non-zero, put labels on each line. The name of each field will be the first 25 characters of each line (approximately as they appear in the HP 3563A or HP 3562A *Programming Manual*). The default is on.

/E:<flag>   If <flag> is non-zero, expand enumerated types to ASCII names (approximately as they appear in the HP 3563A or HP 3562A *Programming Manual*). The default is on.

Example

After copying the capture/throughput file to your PC,using the LIF CP command (LIF CP 702:CSGLITCH GLITCH), type:

DFHDR63 GLITCH GLITCH.ASC

The file GLITCH.ASC will include a list such as the following.

```
Complex_data)_flag        False
Bytes_per_point           2
Points_per_record         2048
Channel_type.             Channel_1
Bandwidth_units           Hertz
X_units                   Hertz
Delay_channel             Channel_2
Delay_count               0
Partial_record            0
Interleave                0
#_of_realtime_records     10
Sectors_per_track         0
Skip_track_[1]            0
```

Discussion

The destination file contains one line for each field in the capture/throughput header. Refer to the HP 3563A or HP 3562A *Programming Manual* for descriptions of the individual fields in the data file. All integers, reals, and long reals will be converted to ASCII, strings will be surrounded by double quotes ("), and enumerated types will be converted to ASCII integers (if /E:0 is specified).

DOWNLOAD

Transfer data over the personal computer's RS-232 port and place it in the destination file.

Syntax   DOWNLOAD <dfile> [/U] [/O] [/B:<baud>] [/P:<port>]

Options

<dfile>   Output file name.

/U   Show help (usage information) for this program.

/O   Overwrite <dfile> if it exists.

/B:<baud>   Select the baud rate for the communications port. Valid baud rates are 110, 150, 300, 600, 1200, 2400, 4800, 9600, or 19200. The default value is 9600.

/P:<port>   Select the communications port number. Valid port numbers are 1 – 4. The default value is 1.

Example

Transfer a data file to a file (FREQ.X32) on a personal computer, using the RS-232 port (port 2).

          DOWNLOAD FREQ.X32/B:19200 /P:2

See also the batch files ''60_SDF.BAT'' and ''69_SDF.BAT'' in appendix C.

HEADER63

Converts the data header portion of a binary data file <sfile> from the HP 3563A (or HP 3562A) to an ASCII form [dfile]. Refer to the HP 3563A or HP 3562A *Programming Manual* for descriptions of the individual fields in the data file. HP 3563A and HP 3562A data files on a LIF file system start with the letters ''DA.''

Syntax   HEADER63 <sfile> [dfile] [/U] [/O] [/L:<flag>] [/E:<flag>]

Options

<sfile>   Input HP 3563A/62A data file name.

[dfile]   Optional output SDF file name. If no file name is specified, output to display.

/U   Show help (usage information) for this program.

/L:<flag>   If <flag> is non-zero, put labels on each line. The name of each field will be the first 25 characters of each line (approximately as they appear in the HP 3563A or HP 3562A *Programming Manual*). The default is on.

/E:<flag>   If <flag> is non-zero, expand enumerated types to ASCII names (approximately as they appear in the HP 3563A or HP 3562A *Programming Manual*). The default is on.

Example

After copying the data file to your PC, using the LIF CP command (LIF CP 702:DAFREQ FREQ1), type:

> HEADER63 FREQ1 FREQ1.ASC /L:0

The file FREQ1.ASC will include a list, such as the following.

> Power_spectrum_1
> 801
> 801
> 1
> Channel_1
> No_channel
> 87
> Freq uency
> RMS
> Volts_squared
> Hertz
> "AUTO MATH"
> ""
> ""
> ""
> True
> Fa lse

Discussion

The destination file contains one line for each field in the data header. Refer to the HP 3563A or
HP 3562A *Programming Manual* for description of the individual fields in the data file.

- All integers, reals, and long reals will be converted to ASCII.
- Strings will be surrounded by double quotes (").
- Enumerated types will be converted to ASCII integers (if /E:0 is specified).

HPIB63

Transfer the active trace data from an HP 3562A/63A over the personal computer's HP-IB and place it in the destination file. Use 63TOSDF.EXE to convert the file to SDF. If /A is not specified, then HP-IB will be scanned for the first instrument that is an HP 3562A or HP 3563A.

Syntax

HPIB63 <dfile> [/U] [/O] [/I] [/A:<hpibAdr>]

Options

<dfile>   Destination HP 3562A/63A file name

/U   Show this usage.

/O   Overwrite <dfile> if it exists.

/I   Show information on all instruments connected to HP-IB.

/A:<hpibAdr>   Specify the HP-IB address of the instrument.

Example

Transfer the active trace to a file (FREQ.63) on a personal computer (let HPIB63 find the instrument) and convert it to ASCII.

                    HPIB63 FREQ.63
                    63TOSDF FREQ.63 FREQ.DAT
                    SDFTOASC FREQ.DAT

See also the batch files ''IB63_ML.BAT'', ''IB63_MX.BAT'', ''IB63_PLT.BAT'', and ''IB63_SDF.BAT'' in appendix C.

REPEAT

Repeatedly calls the specified program, varying the file name (with wild cards) and the destination file name, or all the results/rows/columns/scans in the specified file(s).

Syntax   REPEAT [/U] [/I] [/E] [/A] [<program>] [/D:*] [/R:*] [/C:*] [/S:*] <sfile> [(dfile)]

Options

<program>	Program to REPEAT.
/U	Show this usage.
/I	Ignore errors.
/E	Echo commands before executing program.
/A	Append to destination path.
/D:*	Repeat for all data results.
/R:*	Repeat for all rows in the specified data result(s).
/C:*	Repeat for all columns in the specified data result(s).
/S:*	Repeat for all scans in the specified data result(s).
<sfile>	Source path and file name for SDF file. Repeat for all file(s) by specifying wild card file name.
[(dfile)]	Destination path and file name. It may contain wild card (e.g. *.asc) to match the base name in the <sfile> or pound sign (e.g. #.asc) for sequencing the file name (e.g. 0.asc, 1.asc, etc.) and must have an open/close parentheses. The destination file may be included in an option (e.g. /P: (*.plt) for viewdata).

Example

To convert all the scans in MAP.DAT to ASCII and put each scan in a different file (e.g. 0.asc, 1.asc, etc.):

REPEAT SDFTOASC /S:* MAP.DAT (#.ASC)

Same as above, but append all ASCII data to one file (e.g. map.asc):

REPEAT /A SDFTOASC /S:* MAP.DAT (MAP.ASC)

Plot all results/rows/scans in all *.dat files and put all plotted results for each file in a plot file with the same base name as the data file (e.g. *.plt):

REPEAT /A VIEWDATA /D:* /R:* /S:* *.DAT /P:(*.PLT)

SDFEDIT

Edit (change) header fields in an SDF file. Fields are not case sensitive and may be abbreviated. Multiple fields and Multiple SDF header types may be specified. If no field is specified, then all fields in the header are displayed.

Syntax   SDFEDIT <file> [/U] [/I] [/F] [/M] [/D:<data>] [/V:<vect>] [/C:<chan>] [/S] [/T:<comment>] [/B:<bigScan>] [/N:<scanVar>] [/Q:<unique>] <field>=<value> ...

Options

<file>	SDF file to edit.
/U	Show this usage.
/I	Information only on <file> (i.e. measurement results available).
/F	Edit SDF_FILE_HDR.
/M	Edit SDF_MEAS_HDR.
/D:<data>	Edit specified SDF_DATA_HDR.
/V:<vect>	Edit specified SDF_VECTOR_HDR.
/C:<chan>	Edit specified SDF_CHANNEL_HDR.
/S	Edit SDF_SCAN_HDR.
T:<comment>	Edit specified SDF_COMMENT_HDR
B:<bigScan>	Edit specified SDF_SCAN_BIG (default 0)
N:<scanVar>	Edit specified SDF_SCAN_VAR (default 1)
Q:<unique>	Edit specified SDF_HDR (Unique)
<field>=<value>	Field to change with the specified value. An Enumerated Integer may be specified by its integer value or name.
<field>	Display current value of field.
<field>=	Display type of field. For an Enumerated Integer, this also shows the names associated with the integer values.

Example

To change the point and/or direction on an already existing SDF file, see the following SDFEDIT example:

To change the point number for channel 0 in the file test.dat to point 52 type:

> sdfedit test.dat /c:0 point=52

To see the current point number type:

> sdfedit test.dat /c:0 point

To see the choices for direction for channel 0 type:

> sdfedit test.dat /c:0 direction=

or

> sdfedit test.dat /c:0 dir=

You can abbreviate field names. To set the direction to 'TX__DIRECTION'

> sdfedit test.dat /c:0 direction=TX_DIRECTION

or

> sdfedit test.dat /c:0 dir=tx

or

> sdfedit test.dat /c:0 dir=7

To see all the field names (and current values) in the channel header type:

> sdfedit test.dat /c:0

You can set multiple fields at the same time. To set the point to 5 and the direction to Z_DIRECTION type:

> sdfedit test.dat /c:0 POINT=5 DIRECTION=Z_DIRECTION

or

> sdfedit test.dat /c:0 poi=5 dir=z

This example also shows that the field names and the selection names are NOT case sensitive. You can change fields in more than one header at the same time. To set the point to 2 in the SDF_CHANNEL_HDR and the measTitle to ''this title'', type:

> sdfedit test.dat /m meastitle="this title" /c:0 poi=2

This example also shows that setting a field value that includes blanks requires you to put quotes around the field value.

SDFPRINT

Converts the contents of the input SDF file <sfile> to ASCII and prints all or portions of it. Either the SDF record headers (default) and/or the data associated with the SDF_SCAN_HDR, XDATA, YDATA, and comment records may be printed.

Syntax SDFPRINT <sfile> [dfile] [/U] [/O] [/I] [/V] [/H] [/T:<type>] [/A] [N:<#perLine>]
[/D:<data>] [/S:<scan>] [/R:<row>] [/C:<col>] [/P:<flag>] [/E:<flag>] [/W:<flag>]

Options

<sfile> Input file name for SDF file.

[dfile] Optional output file name for ASCII information (default is the display).

/U Show help (usage information) for this program.

/O Overwrites [dfile] if it exists.

/I Show information only on <sfile>, including data result number, result name, the number of rows, columns, scans available, the number of data points, whether the data is real or complex, and type of X spacing between data points.

/V Validate SDF file (does consistency check).

/H Print just the SDF record headers (default).

/T:<type> Print data for SDF SCAN (/T:S), X (/T:X), Y (/T:Y) data, unique records (/T:U), or comment records (/T:C). If /T is specified, all data is printed.

/A Print both headers and data.

/N:<#perLine> Numbers of data for each printed line. The default is 1.

/D:<data> Print data only for the specified data result (see figure 2-1). If not specified, then all results in <sfile> will be selected.

/S:<scan> Print data only for the specified scan (see figure 2-3). If not specified, then all scans in <sfile> will be selected.

/R:<row> Print data only for the specified row (see figure 2-2). If not specified, then all rows in <sfile> will be selected.

/C:<col>   Print data only for the specified column. If not specified, then all columns in <sfile> will be selected.

/P:<flag>   If <flag> is non-zero, print SDF record titles (default on).

/E:<flag>   If <flag> is non-zero, expand enums to labels (default on).

/W:<flag>   If <flag> is non-zero, scale the data for the window and engineering units. (default on).

Example

To find out what results are available in an SDF measurement results file from HP 3566A/67A (swept sine measurement):

SDFTO63 /I SINEARB.DAT

This will produce a list on the display such as the following.

Data	Name	Rows	Cols	Scans	Points	Complex	Space
0	Freq Response				103		Arb
1	Nrml Variance	2			103		Arb
2	Input Power Ch1				103		Arb
3	Output Power Ch2				103		Arb
4	Cross Power				103		Arb

This example shows 5 data results (0 thru 4) are available with 103 arbitrarily spaced points (indicative of swept sine data, see figure 2-1).

The ''Rows'' field (if present) is usually associated with the number of channels of data available for single channel results (see figure 2-2). For example, 2 channels of ''Normalized Variance'' measurement results are present.

At this time, the ''Columns'' field is not used (there is always only 1 column of data) by any instrument using SDF as a data format.

The ''Scans'' field indicates that the data contains Waterfall results from HP 3566A/67 (see figure 2-3).

To print all the headers to this SDF file type:

> SDFPRINT SINEARB.DAT

To print only the data for ''Normalized Variance'' channel 2 (row 0 is channel 1, row 1 is channel 2) type:

> SDFPRINT SINEARB.DAT /D:1 /R:1

To now print this data in pure ASCII with no record headers:

> SDFPRINT SINEARB.DAT /D:1 /R:1 /P:0

Discussion

The HP 3566A/67A generally puts all results from a measurement in a single file (see example above).

The following types of records are present in an SDF file:

Name	Description
SDF_FILE_HDR	Provides an index to the file.
SDF_MEAS_HDR	Contains settings of measurement parameters.
SDF_DATA_HDR	Information on one type of measurement result (X/Y info). One measurement result can contain many rows (e.g. channels) of data results.
SDF_VECTOR_HDR	Information on which channels were used for each data result. There is one SDF_VECTOR_HDR for each row (& column) of data in a measurement result. Each SDF_VECTOR_HDR may reference up to 2 channels.
SDF_CHANNEL_HDR	Contains channel-specific information for one channel used in a measurement.
UNIQUE record	Contains instrument-specific information.
SDF_SCAN_STRUCT	Tells you how vectors are organized in the Y-axis Data record when the measurement includes multiple scans of data.
SDF_SCAN_BIG	Tells you how vectors are organized in the Y-axis Data record when the measurement includes a large number of scans of data.
SDF_SCAN_VAR	Information about each scan. This is for measurements that include multiple scans of data.

SDF_COMMENT_HDR   Contains comments about the data.

XDATA   Contains the X-axis data needed to reconstruct a trace. This can be used
                         by Swept Sine or Order Analysis measurements.

YDATA   Contains the Y-axis data needed to reconstruct any trace.

SDFTEXT

Modify/print SDF file <sfile> comment text data. If [cfile] is specified then the comment in the SDF file is modified, otherwise the comment is printed. If the /A option is not specified, then the previous comment in the file is replaced with [cfile]. If there is not enough space in the specified comment for the text, then the text will be truncated.

If no space was allocated for the comment area by the instrument us SDFTOSDF with the /M option to create a comment area in the SDF file.

Syntax

SDFTEXT <sfile> [cfile] [/U] [/T:<comment>] [/A]

Options

<sfile>   SDF file.

[cfile]   Input ASCII comment file.

/U   Show this usage.

/T:<comment>   Select specified SDF_COMMENT_HDR (default 0).

/A   Append text to end of comment in file.

Example

SAMPLE.TXT contains some ASCII text.

SDFTEXT TIME.DAT SAMPLE.TXT /A

This command appends the contents of SAMPLE.TXT to the end of the TIME.DAT SDF file comment area.

SDFTO58

Convert SDF file <sfile> to the Data Set 58 format. The result is placed in an output file [dfile] if specified; otherwise the result is sent to the screen.

Syntax   SDFTO58 <sfile> [dfile] [/U] [/O] [/I] [/A] [/H] [/D:N1[-N2]] [/R:N1[-N2]] [/C:N1[-N2]] [/S:N1[-N2]]

Options

<sfile>   Input SDF file name.

[dfile]   Optional output file name for ASCII information. If no file name is specified, output to display.

/U   Show help (usage information) for this program.

/O   Overwrite [dfile] if it exists.

/I   Show information only on <sfile>, including data result number, result name, the number of rows, columns, scans available, the number of data points, whether the data is real or complex, and type of X spacing between data points.

/A   Convert all frequency lines (normally the lines likely to be affected by aliasing are not converted)

/H   Do not output header records (allows you to output an array of data only)

/D:N1[-N2]   Convert the data at data number N1 (to data number N2). Range; 0 to n–1.

/R:N1[-N2]   Convert the data at row N1 (to row N2). Range; 0 to n–1.

/C:N1[-N2]   Convert the data at column N1 (to column N2). Range; 0 to n–1.

/S:N1[-N2]   Convert the data at scan N1 (to scan N2). Range; 0 to n–1.

SDFTO63

Converts a single measurement result from an SDF data file <sfile> to an HP 3563A (or HP 3562A) data file <dfile>. HP 3563A/62A data files on a LIF system start with the letters ''DA.''

Syntax   SDFTO63 <sfile> <dfile> [/U] [/O] [/I] [/D:<data>] [/S:<scan>] [/R:<row>] [/C:<col>]

Options

<sfile>	Input SDF file name.
<dfile>	Output file name for HP 3563A/62A data file. The file name must begin with DA, for example DATRACE1, if you plan to use LIF CP on the file.
/U	Show help (usage information) for this program.
/O	Overwrite <dfile> if it exists.
/I	Show information only on <sfile>, including data result number, result name, the number of rows, columns, scans available, the number of data points, whether the data is real or complex, and type of X spacing between data points.
/D:<data>	Select data result for the specified SDF_DATA_HDR. If not specified, then the first result (result 0) in <sfile> will be selected. Use /I to see a list of the result names and numbers.
/S:<scan>	Select data for the specified scan (map line or waterfall step, see figure 2-3). If not specified, then the first scan (scan 0) in <sfile> will be selected.
/R:<row>	Select data for the specified row (see figure 2-2). If not specified, then the first row (row 0) in <sfile> will be selected.
/C:<col>	Select data for the specified column. If not specified, then the first column (column 0) in <sfile> will be selected. At this time, the ''Columns'' field is not used (there is always only 1 column of data) by any instrument using SDF as a data format.

Example 1

To find out what results are available in an SDF measurement results file from HP 3566A/67A swept sine measurement:

SDFTO63 /I SINEARB.DAT

This will produce a list on the display in the following form:

Data	Name	Rows	Cols	Scans	Points	Complex	Space
0	Freq Response				103		Arb
1	Nrml Variance	2			103		Arb
2	Input Power Ch1				103		Arb
3	Output Power Ch2				103		Arb
4	Cross Power				103		Arb

This example shows 5 data results (0 thru 4) are available with 103 arbitrarily spaced points (indicative of swept sine data, see figure 2-1).

The ''Rows'' field (if present) is usually associated with the number of channels of data available for single channel results (see figure 2-2). For example, 2 channels of ''Normalized Variance'' measurement results are present.

At this time, the ''Columns'' field is not used (there is always only 1 column of data) by any instrument using SDF as a data format.

The ''Scans'' field indicates that the data contains Waterfall results from HP 3566A/67 (see figure 2-3).

Example 2

To convert the ''Normalized Variance'' channel 2 (row 0 is channel 1, row 1 is channel 2) to HP 3563A format, perform the following steps:

1. Convert the file to Linear spaced data (SDFTOSDF).

   SDFTOSDF SINEARB.DAT SINELIN.DAT /P (convert to linear spacing)

2. Extract the result with SDFTO63.

   SDFTO63 SINELIN.DAT VAR2.63 /D:1 /R:1 (/R:1 is channel 2)

3. Copy the HP 3563A data file to a LIF formatted disk (to be read by HP 3563A):

   LIF CP VAR2.63 B:DAVAR2 (HP 3563A data files start w/ ''DA'')

Discussion

Only one result may be converted into HP 3563A data file format. If no options are selected that specify which data to convert, then the first result is selected. The HP 3563A puts one measurement result in a data file, whereas the HP 3566A/67A generally puts all results from a measurement in a single file (see previous example).

SDFTO63 can only convert data that is linear or log spaced with up to 801 lines of resolution, 2048 real points, or 1024 complex points. To get arbitrary spaced data to the HP 3563A, first convert the SDF data to evenly spaced data (linear or log) with SDFTOSDF. To get data with more than 2048 real / 1024 complex points to the HP 3563A, use SDFTOSDF to interpolate the SDF data file to fewer number of points before converting with SDFTO63.

SDFTOASC

Convert an SDF data file to an ASCII data file and place the result in the destination file (if it is entered). Otherwise view the result on the screen.

Note   All header information is lost when you convert to ASCII format.

☞

SDFTOASC <sfile> [dfile] [/U] [/O] [/I] [/A] [/X] [/T:<c1>,<c2>] [/Y:<units>]
[/M:<dBmRef>] [/G:<dBRef>] [/P:<points>] [/D:<data>] [/R:<row>[-<rowEnd>][,C]]
[/C:<col>] [/S:<scan>[−<scanEnd>][,C]] [/L] [/F:<format>] [/B:<string>]

<sfile>   Input SDF file.

[dfile]   Output ASCII file.

/U   Show help (usage information) for this program.

/O   Overwrite [dfile] if it already exists.

/I   Information only on the <sfile>.

/A   All the frequency lines.

/X   Include X data in the file.

/T:<c1>,<c2>   Y coordinates, <c1> or <c2> may be one of the following:
                    B=dBm, D=dB, S=dB signed, M=Mag, R=Real, I=Imag, P=Phase, U=Unwrap
                    phase. Default is R, I for complex data, R for real data.

/Y:<units>   Y units, one or more of the following:
                    L=Linear, P=Power, D=Density, R=RMS, A=Radians

/M:<dBmRef>   dBm impedance reference. Default is use input impedance from data file if
                    < 1 MΩ (else use 50Ω).

/G:<dBRef>   Magnitude value used as a reference for dB type of coordinates. If /G with no
                    parameter, then default is 20E-6 (dBSPL).

/P:<points>   Number of points per line. The default is 1.

/D:<data>   Select data for the specified SDF_DATA_HDR. The default is 0.

/R:<row> [-<rowEnd>] [,C]	Select data for the specified row range. The default is 0. If ''C'' is specified, then the row data will be arranged as one row per column.
/C:<col>	Select data for the specified column. The default is 0.
/S:<scan> [-<scanEnd>] [,C]	Select data for the specified scan range. The default is 0 (1 scan). If ''C'' is specified, then the scan data will be arranged as one scan per column.
/L	Orient the output data in lines instead of columns.
/F:<format>	C printf format string. The default is ''%14.6le.'' This specifies a double (long float) with a width of 14 columns and a precision of 6 in exponential format. Other formats are:

$\quad$ %lf — regular floating point format
$\quad$ %le — exponential floating point format (also %1E)
$\quad$ %lg — regular or exponential, whichever is shorter (also %1G)

/B:<string>	Field separator string. The default is '' '' (space.) Viewdata requires commas between real and imaginary data.

Example 1

Convert an SDF data file to an ASCII data file:

$\qquad$ SDFTOASC TEST.DAT TEST.TXT

Example 2

Convert an SDF data file to an ASCII data file using a comma ('','') as the field separator:

$\qquad$ SDFTOASC TEST.DAT TEST.TXT /B:,

Example 3

Convert a spectrum SDF data file to ASCII (converting the data to Vrms^2 /Hz (PSD)).

$\qquad$ SDFTOASC SPEC.DAT SPEC.TXT /Y:PRD

The ''P'' converts the data to V^2 (if not already power data), the ''R'' converts the data to rms, and the ''D'' converts the data to density units (/Hz).

Example 4

Convert the complex data to dB, phase (instead of the default real, imaginary), and phase is in radians.

    SDFTOASC FRF.DAT FRF.TXT /T:D,P /Y:A

Example 5

Convert a waterfall to a format which a spreadsheet can read in as 1 column per spectrum.

    SDFTOASC WFAL.DAT WFAL.TXT /S:0−1000,C

If the ending scan that is selected is too large for the waterfall, then the ending scan is set to the last scan in the waterfall. If the /L option is specified, then each spectrum is in a row (line) rather than in a column.

Example 6

Convert a sound pressure spectrum to dBSPL.

    SDFTOASC NOISE.DAT NOISE.TXT /T:D /Y:R /G:20E−6

The units of the NOISE.DAT file are Pascals, so the /T:D option converts to dB, and the /G:20E−6 option sets the dB reference to 20 micro-Pascals.

SDFTOBIN

Convert the source SDF file to the destination binary file. The binary data has the bytes swapped in the same direction as the rest of the DOS environment.

Syntax   SDFTOBIN <sfile> <dfile> [/U] [/O] [/I] [/A] [/X] [/T:<c1>,<c2>] [/Y:<units>] [/M:<dBmRef>] [/G:<dBRef>] [/D:<data>] [/R:<row>] [/C:<col>] [/S[:<scan>[− <scanEnd>]]] [/F:<format>]

Options

<sfile>	Input SDF file.
<dfile>	Output binary file.
/U	Show this usage.
/O	Overwrite <dfile> if it exists.
/I	Info only on <sfile> (that is, measurement results available).
/A	All frequency lines.
/X	Include X data in the file.

/T:<c1>,<c2>   Y coordinate, <c1> or <c2> may be one of the following:

> B=dBm
> D=dB
> S=dB signed
> M=Mag
> R=Real
> I=Imag
> P=Phase
> U=Unwrap phase

Default is R,I for complex data; R for real data.

/Y:<units>   Y units, one or more of the following:

> L=Linear,
> P=Power,
> D=Density,
> R=RMS,
> A=Radians

/M:<dBmRef>]   dBm impedance reference. Default is use input impedance from the data file if 1 MOhm (else use 50 Ohm).

/G:<dBRef>]   Magnitude value used as a reference for dB type of coordinates. If /G with no
                     parameter, then default is 20E–6 (dBSPL).

/D:<data>]   Select data for the specified SDF_DATA_HDR (default is 0).

/R:<row>]   Select data for the specified row (default is 0).

/C:<col>]   Select data for the specified column (default is 0).

/S[:<scan>   Select data for the specified scan range (default is 0).
 [–<scanEnd>]]

/F:<format>   Binary format, one of the following:
                                F=32-bit float
                                D=32-bit float

Example

The file TIME.DAT contains 2048 real time points.

> SDFTOBIN TIME.DAT TIME.BIN /F:F

This command creates a binary file TIME.BIN with 2048 floats. Since each float is 4 bytes, the file is
2048 * 4 = 8192 bytes long.

SDFTOML

Converts data from an SDF format file to the format used in the PC-MATLAB program. The created file or files have the suffix ''.MAT'' if no destination file is specified.

If the batch file naming convention is used or an output file is specified, then only one measurement result is translated. The batch file naming convention clears the <sfile> suffix and adds a MAT suffix.

Syntax   SDFTOML <sfile> [/U] [/O] [/I] [/A] [/R] [/X] [/Q] [/B] [/D:<data>] [/R:<row>] [/C:<col>]
[/S:<scans>] [/Y:<units>]

Options

<sfile>	Input SDF file name.
/U	Show help (usage information) for this program.
/O	Overwrites <PC-MATLAB file> if it exists.
/I	Show information only on <sfile>, including data result number, result name, the number of rows, columns, scans available, the number of data points, whether the data is real or complex, and type of X spacing between data points.
/A	Convert all frequency lines (normally the lines likely to be affected by aliasing are not converted).
/R	Output the PC-MATLAB file matrices in a row instead of a column.
/X	Output the X-axis data points.
/Q	Query for partial conversion.

For information on naming conventions, refer to ''Naming Conventions for Matrix Files'', ''Naming Conventions for Y-axis Variables'' and ''Naming Conventions for X-axis Variables'' under SDFTOMX.

/B	Use batch file naming convention (see appendix C).
/D:<data>	Extract data only for the specified data header (default all).
/R:<row>	Extract data only for the specified data row (default all rows).
/C:<col>	Extract data only for the specified data column (default all columns).

/S:<scans>   Extract data only for the specified data scans (default all scans). For example: /S:0
                       means scan 0 only, /S:0-4 means scans 0 through 4.

/Y:<units>   Y units, one or more of the following:
                       L=Linear, P=Power, D=Density, R=RMS

See the batch files ''60_ML.BAT'' and ''63_ML.BAT'' in appendix C.

Example

Convert a spectrum SDF data file to PC-Mathlab file. (converting data to Vrms^2/Hz (PsD))

                       SDFTOML SPEC.DAT /Y:PRD

The ''P'' converts the data to V^2 (if not already power data), the ''R'' converts the data to rms, and the
''D'' converts the data to density units (/Hz).

SDFTOMX

Converts data from the SDF format to the format used in the MATRIXx program. The created file or files have the suffix ''.MAT'' if no destination file is specified.

If the batch file naming convention is used or an output file is specified, then only one measurement result is translated. The batch file naming convention clears the <sfile> suffix and adds a MAT suffix.

Syntax SDFTOMX <sfile> [dfile] [/U] [/O] [/I] [/A] [/R] [/X] [/Q] [/B] [/D:<data>] [/R:<row>] [/C:<col>] [/S:<scans>] [/Y:<units>]

Options

<sfile>	Input SDF file.
[dfile]	Optional output MATRIXx file.
/U	Show help (usage information) for this program.
/O	Overwrite <MATRIXx file> if it exists.
/I	Show information only on <sfile>, including data result number, result name, the number of rows, columns, scans available, the number of data points, whether the data is real or complex, and type of X spacing between data points.
/A	Convert all frequency lines (normally the lines likely to be affected by aliasing are not converted).
/R	Output the MATRIXx file matrices in a row instead of a column.
/X	Output the X-axis data points.
/Q	Query for partial conversion.
/B	Use batch file naming convention (see appendix C).
/D:<data>	Extract data only for the specified data header (default all).
/R:<row>	Extract data only for the specified data row (default all rows).
/C:<col>	Extract data only for the specified data column (default all columns).

/S:<scans> Extract data only for the specified data scans (default all scans). For example: /S:0 means scan 0 only, /S:0-4 means scans 0 through 4.

/Y:<units> Y units, one or more of the following:
L=Linear, P=Power, D=Density, R=RMS

See the batch files ''60_MX.BAT'' and ''63_MX.BAT'' in appendix C.

Example

Convert a spectrum SDF data file to a MATRIXx file (converting data to Vrms^2/Hz (PSD)).

SDFTOMX SPEC.DAT /Y:PRD

The ''P'' converts the data to V^2 (if not already power data), the ''R'' converts the data to rms, and the ''D'' converts the data to density units (/Hz).

Naming Conventions for Matrix Files

If the destination file is not specified and the /B option is not specified, then the output filenames are derived from the measurement result names used in the analyzer. The extension .MAT is added. For example, frequency response data is saved to a file called FREQRESP.MAT. For data which includes more than one measurement result a separate file is used for each result. For example Histogram data is saved to HISTOGRA.MAT, PDF.MAT, and CDF.MAT.

If the destination file or the /B option is specified, then the only one result will be converted.

Naming Conventions for Y-axis variables

Within the file, the individual trace data is saved to matrix variables with unique names. The names are dependent on the channels and the map lines (if there are more than one) that comprise the trace.

The variable naming format is shown in the following table:

Y-Axis Variable	Notes
c<N1>:	Where N1 is the channel number for data that is derived from a single channel or 0 in the case of no channel.
c<N1>m<N2>:	Where N1 is the channel number and N2 is a map line number in the case of multiple map lines.
o<N1>i<N2>:	Where N1 is the output channel number and N2 is the input channel number for data that is derived from two channels.
o<N1>i<N2>m<N3>:	Where N1 is the output channel number, N2 is the input channel number, and N3 is a map line number in the case of multiple map lines.

Examples of variable names:

Variable	File	Notes
c2	INSTTIME.MAT	Channel 2 of an Instantaneous Time measurement.
c2m3	POWERSPE.MAT	The third map line in channel 2 of a Power Spectrum measurement.
o2i1	FREQRESP.MAT	Output channel 2 versus input channel 1 of a Frequency Response measurement.

These naming conventions were chosen to ensure that each variable name in the file is unique. You can change the names in MATRIXx or PC-MATLAB if you wish. For example you can create names which are easier to remember or which more clearly describe the data.

Naming Conventions for X-axis Variables

If you choose the /X option the program does will create a variable of the form:

$$<name>x$$

for each Y-axis variable. <name> is the Y-axis variable name as described in the table above.

If you do not choose the /X option the program will create the following three variables:

Variable	Description
<name>x0	the starting X value
<name>xi	the linear X increment between points
<name>xl	the log X increment between points

The variables <name>x0, <name>xi, and <name>xl can be used to calculate the X-axis data as follows:

$$X = (Xlast + <name>xi) * <name>xl$$

where Xlast is the previous point's X value and the X value for the first point is <name>x0.

SDFTOSDF

Extracts a subset of results from an SDF file <sfile> and creates another SDF file <dfile>. The type of spacing and number of points for the results may be modified through resampling using polynomial or spline interpolation. (See chapter 2.)

Syntax   SDFTOSDF <sfile> <dfile> [/U] [/O] [/I] [/D:<data>] [/S:<scans>] [/R:<row>] [/C:<col>] [/E:<pts>] [/P:<pts>] [/L:<pts>] [/T:<type>] [/N:<pts>] [/X:<first>,<last>] [/B:<rec>] [/G:<gap>] [/A:<add>] [/F:<form>,<data>] [/M:[cfile] [,<comSz>]] [/V] [/Q:[uniq]]

Options

<sfile>   Source path and file name for SDF file.

<dfile>   Destination path and file for SDF file.

/U   Show help (usage information) for this command.

/O   Overwrites <dfile> if it exists.

/I   Show information only on <sfile>, including data result number, result name, the number of rows, columns, scans available, the number of data points, and type of X spacing between data points.

/D:<data>   Extract data only for the specified data result (see figure 2-1). If not specified, then all results in <sfile> will be selected.

/S:<scans>   Extract data only for the specified scan (see figure 2-3). If not specified, then all scans in <sfile> will be selected. Either a single scan (e.g. /S:0 for scan 0) or a range of scans (e.g. /S:0-4 for scans 0 thru 4) may be selected.

/R:<row>   Extract data only for the specified row (see figure 2-2). If not specified, then all rows in <sfile> will be selected.

/C:<col>   Select column within result. If not specified, then all columns in <sfile> will be selected.

/E:<pts>   Interpolate to the specified number of lin/log evenly spaced points. The X data is scanned to determine if the data is approximately linear or log spaced. If the number of points is not specified, then the original number of points in the data result is maintained. This is useful if you want to do math or curve fit swept sine data, or examine results with a large number of lines of resolution on an analyzer that is limited to fewer lines.

/P:\<pts>	Interpolate to the specified number of linear spaced points. If the number of points is not specified, then the original number of points in the data result is maintained. This is useful if you want to do math or curve fit swept sine data, or examine results with a large number of lines of resolution on analyzer that is limited to fewer lines.
/L:\<pts>	Interpolate to the specified number of log spaced points. If the number of points is not specified, then the original number of points in the data result is maintained. This is useful if you want to do math or curve fit swept sine data, or examine results with a large number of lines of resolution on analyzer that is limited to fewer lines.
/T:\<type>	Type of interpolation: S = Spline, P = Polynomial. Default is P.
/N:\<pts>	Number of points for polynomial interpolation. Range: 2 – 10. Default is 2. The default spacing is linear (polynomial interpolation and 2 points).
/X:\<first>, \<last>	First and last X value of data (default is all the data). Can specify either or both.
/B:\<rec>	Bar record from being copied: U = Unique
/G:\<gap>	Add the specified # of bytes between each SDF record type area
/A:\<add>	Add the specified # of bytes to each SDF record
/F:\<form>, \<data>	Convert number format of y data for the specified data header to the format specified: S=SHORT16, L=LONG32, F=FLOAT32, D=DOUBLE64, Data # is optional. If the data number is not specified, then all results are converted.
/M:[cfile] [,\<comSz>]	Add a comment record and/or reserve comSz bytes for a comment record (default is 1024)
/V	Add scan var duplicate of scan structure
/Q:[uniq]	Add the specified unique record: M = Modulation

Example 1

To find out what results are available in an SDF measurement results file from HP 3566A/67A (swept sine measurement):

> SDFTOSDF /I SINEARB.DAT

This will produce a list on the display such as the following.

Data	Name	Rows	Cols	Scans	Points	Complex	Space
0	Freq Response				103		Arb
1	Nrml Variance	2			103		Arb
2	Input Power Ch1				103		Arb
3	Output Power Ch2				103		Arb
4	Cross Power				103		Arb

This shows 5 results are available (0 thru 4) with 103 arbitrarily spaced points (indicative of swept sine data). The ''Rows'' field (if present) if usually associated with the # of channels of data available for single channel results.

For example, 2 channels of ''Normalized Variance'' measurement results are present. At this time, the ''Columns'' field is not used (i.e. always only 1 column of data) by any instrument using SDF as a data format. The ''Scans'' field indicates that this data contains Waterfall results from the HP 3566A/67.

To extract and convert the ''Normalized Variance'' channel 2 (row 0 is channel 1, row 1 is channel 2) to 401 linear spaced points:

> SDFTOSDF SINEARB.DAT SINELIN.DAT /D:1 /R:1 /P:401

Example 2

To extract and convert an FRF from the HP 3563A (801 lines at 0 to 100 kHz) to the HP 35665A (401 lines at 0 to 102.4 kHz):

> 63TOSDF FRF.63 FRF63.DAT
> SDFTOSDF FRF63.DAT FRF665.DAT /P:401 /X:,102400

Discussion

The HP 3566A/67A generally puts all results from a measurement in a single file (see example above).

If no options are selected that specify which data to convert, then all results are selected. This has the effect of copying everything in the source SDF file <sfile> to the destination SDF file <dfile> (and resampling all data results, if specified). This also has the effect of converting SDF Level 1 files (previously used HP 3566A/67A) to the newer SDF Level 2 files (now used by HP 3566A/67A).

Data is interpolated only if either the /E, /P, or /L option is specified.

The ability to convert arbitrarily space data (Swept Sine measurements) to linear or log space data (through resampling) is necessary when importing data to the HP 3563A.

The ability to change the number of points in a data result (through resampling) is required when importing data into an instrument that supports limited data sizes. For example, the HP 35660A supports up to 1024 complex points in a data result, whereas the HP 3566A/67A can generate results up to 16 K real points.

Spline interpolation (/T:S) is useful when converting from a smaller number of points to a much larger number of points when you want the data to be smoothed. Polynomial interpolation (/T:P) with linear fit (/N:2) or quadratic fit (/N:3) is faster than a spline fit and is useful when converting to nearly the same number or fewer points. (Quadratic fit is also good for interpolating to a larger number of points.)

If the data is real, the real part of the data is interpolated. If the data is complex, the magnitude and phase parts of the data are interpolated separately.

With the /X option, you may specify X values either within the data or outside of the data. If an X value specified is outside of the data, the curve is extrapolated to fill in points to the specified X value. For log data, you must specify X values greater than 0.

SDFYDATA

Modify/print SDF file <sfile> Y data. If [afile] or /Y option is specified then the Y data in the SDF file is modified, otherwise the data is printed.

Syntax	SDFYDATA <sfile> [afile] [/U] [/I] [/D:<data>] [/S:<scan>] [/R:<row>] [/C:<col>] [/O:<ofst>[,<unit>]] [/L:<len>[,<unit>]] [/Y:<real>[,<imag>]]
<sfile>	SDF file.
[afile]	Input ASCII file with numbers.
/U	Show this usage.
/I	Info only on <sfile> (that is, measurement results available).
/D:<data>	Select data only for the specified SDF_DATA_HDR (default is 0).
/S:<scan>	Select data only for the specified scan (default is 0).
/R:<row>	Select data only for the specified row (default is 0).
/C:<col>	Select data only for the specified column (default is 0).
/O:<ofst> [,<unit>]	Offset into the data by <ofst>. <unit> may be R (records), P (points) or S (seconds); default is P.
/L:<len> [,<unit>]	Length of data to change, default is one point. Only used if [afile] is not specified.
/Y:<real> [,<imag>]	Value to change SDF file with. Only used if [afile] is not specified.

Example

Print out 10 mS of data after the 2 seconds of data in a time capture file (CAPT.TIM)

> SDFYDATA CAPT.TIM /O:0.01,S /L:2,S

Zero out the same region of data (just printed) in the time capture file.

> SDFYDATA CAPT.TIM /O:0.01,S /L:2,S /Y:0

Copy the first 10 mS of data to the second 10 mS of data in a time capture file.

> SDFYDATA CAPT.TIM > TEMP /L:0.01,S
> SDFYDATA CAPT.TIM TEMP /O:0.01,S

SETUP63

Converts a binary setup state file from the HP 3563A (or HP 3562A) to an ASCII form (written to destination file). Refer to the HP 3563A or HP 3562A programming manual for descriptions of the individual fields in the setup state file. HP 3563A/62A setup state files on a LIF file system start with the letters ''SE.''

Syntax   SETUP63 <sfile> [dfile] [/U] [/O] [/L:<flag>] [/E:<flag>]

Options

<sfile>   Input HP 3563A/62A setup state file name.

[dfile]   Optional output ASCII file name. If no file specified, output to display.

/U   Show help (usage information) for this program.

/O   Overwrite [dfile] if it exists.

/L:<flag>   If <flag> is non-zero, put labels on each line. The name of each field will be the first 25 characters of each line (approximately as they appear in the HP 3563A or HP 3562A *Programming Manual*). The default is on.

/E:<flag>   If <flag> is non-zero, expand enumerated types to ASCII names (approximately as they appear in the HP 3563A or HP 3562A *Programming Manual*). The default is on.

Example

After copying your HP 3563A/62A file from LIF to DOS using the LIF CP command
(LIF CP A:SEFREQ FREQ), type:

SETUP63 FREQ FREQ.ASC /E:0

FREQ.ASC will contain a list of setup parameters. For example:

Measurement_mode   0
Measurement_1   2
Measurement_2   5
Window_type   11
Force/Expon_window_1   0
Force/Expon_window_2   1
Average_type   10
Overlap_percentage   0
Number_of_averages   10
Sweep_#_of_averages   1
Trigger_type   18

To interpret this data, look in the HP 3563A or HP 3562A *Programming Manual* at the table
''Enumerated Types for Instrument State Values'' or omit the /E:0 option.

Discussion

The destination file contains one line for each field in the setup state (except qualifiers, which contain the
''bit'' and ''don't care'' fields on one line). Refer to the HP 3563A or HP 3562A programming manuals
for a description of the individual fields in the setup state file.

- All integers, reals, and long reals will be converted to ASCII.
- Strings will be surrounded by double quotes (").
- Enumerated types will be converted to ASCII integers (if /E:0 is specified).
- Qualifiers will be converted to a series of 1, 0, or X.

SETUP88

Converts a binary setup state file from the HP 3588A to an ASCII form written to a destination file.

Syntax   SETUP88 <sfile> [dfile] [/O] [/U] [/M <mfile>] [/L:<flag>] [/E:<flag>]

Options

<sfile>   Input HP 3588A setup state path and file name.

[dfile]   Optional destination path and file name. If no file is specified, it outputs to the display.

/O   Overwrite [dfile] if it already exists.

/U   Show help (usage information) for this program.

/M   Read HP 3588A math file for additional information.

<mfile>   HP 3588A math path and file name.

/L:<flag>   If <flag> is non-zero, put labels on each line. The name of each field will be the first 25 characters of each line. The default is on.

/E:<flag>   If <flag> is non-zero, expand enumerated types to ASCII names. The default is on.

Example

Copy a setup state file from the HP 3588A LIF formatted disk and convert it to ASCII format:

> LIF CP B:STATE1 STATE1.88
> SETUP88 STATE1.88 STATE1.ASC

State1.ASC will contain a list of setup parameters. The following is an example list.

```
Measurement_mode      Swept Spectrum
Low_distortion        Off
Oversweep             On
Peak_detector         On
 Center_frequency       7.505e+007
Span               1.499e+008
Start_frequency       100000
Stop_frequency        1.5e+008
Manual_frequency      7.505e+007
Frequency_step_size    1000
Frequency_step_mode    Auto
Signal_track          Off
Resolution_BW          17000
Video_BW               26248
Video_filter          Off
etc.  +
```

Discussion

- All integers, reals, and long reals will be converted to ASCII.
- Strings will be surrounded by double quotes (").
- Enumerated types will be converted to ASCII integers (if /E:0 is specified).

SETUP89

Converts a binary setup state file from the HP 3589A to an ASCII form written to a destination file.

Syntax   SETUP89 <sfile> [dfile] [/O] [/U] [/M <mfile>] [/L:<flag>] [/E:<flag>]

Options

<sfile>   Input HP 3589A setup state path and file name.

[dfile]   Optional destination path and file name. If no file is specified, it outputs to the display.

/O   Overwrite [dfile] if it already exists.

/U   Show help (usage information) for this program.

/M   Read HP 3589A math file for additional information.

<mfile>   HP 3589A math path and file name.

/L:<flag>   If <flag> is non-zero, put labels on each line. The name of each field will be the first 25 characters of each line. The default is on.

/E:<flag>   If <flag> is non-zero, expand enumerated types to ASCII names. The default is on.

Example

Copy a setup state file from the HP 3589A LIF formatted disk and convert it to ASCII format:

                        LIF CP B:STATE1 STATE1.89
                         SETUP89 STATE1.89 STATE1.ASC

State1.ASC will contain a list of setup parameters. The following is an example list.

        Measurement_mode   Swept Spectrum
        Low_distortion   Off
        Oversweep   Off
        Peak_detector   Off
        Center_frequency   7.505e+007
        Span   1.499e+008
        Start_frequency   100000
        Stop_frequency   1.5e+008
        Manual_frequency   0
        Frequency_step_size   1000
        Frequency_step_mode   Auto
        Signal_track   Off
        Resolution_BW   290
        Video_BW   26248
        Video_filter   Off
        etc. +

Discussion

- All integers, reals, and long reals will be converted to ASCII.
- Strings will be surrounded by double quotes (").
- Enumerated types will be converted to ASCII integers (if /E:0 is specified).

SOFTCOPY

Transfers data from the HP-IB (IEEE-488 bus) to a file. Allows your PC to emulate a hardcopy device such as a plotter or printer, so screen dumps from various instruments can be captured. Hardcopy from HP-IB Instrument can be routed to an RS-232 printer or plotter by specifying PRN for the output file. Requires an HP 82335A interface card, or National Instrument GPIB-PC interface card.

Syntax   SOFTCOPY [dfile] [/U] [/O] [/L] [/A:<addr>]

Options

[dfile]   Output file.

/U   Show this usage

/O   Overwrite [dfile] if it exists

/L   Longer end-of-data timeout (30 sec). Default is 3 sec.

/A:<addr>   Set primary IEEE 488 bus address. Default is 5, use 31 to emulate a listen-only plotter or printer.

Example 1

To capture a plot from a HP 3562A Dynamic Signal Analyzer in a file, connect the analyzer to the PC's HP-IB card. Then setup the analyzer "HP-IB fctn" as "system controller", and set "plot address" to 5. Start softcopy with the following command

            SOFTCOPY APLOT.PLT /A:5

Then press "start plot" on the analyzer. When the plot is done, the file can be inserted into a Microsoft Word for Windows document.

Example 2

To send a screen dump from an HP 54502A Oscilloscope to a DeskJet printer, connect the Oscilloscope to the PC's HP-IB card. Connect the printer to the PC's parallel interface. Setup the Oscilloscope's "Utility / HP-IB" as "talk only". Start softcopy with the following command:

            SOFTCOPY PRN /O /A:31

Then press the "hardcopy" button on the Oscilloscope.

SYNTH63

Converts a binary synthesis file from the HP 3563A or HP 3562A to an ASCII file, a PC-MATLAB file, or a MATRIXx file. HP 3563A/62A synthesis files on a LIF file system start with the letters ''ST'' (Z-domain ''ZT''); curve fit files start with the letters ''CV'' (Z-domain ''ZF'').

Syntax   SYNTH63 <sfile> <dfile> [/U] [/O] [/L] [/M] [/R]

Options

<sfile>   Input HP 3563A/62A synthesis file name.

<dfile>   Output file name.

/U   Show help (usage information) for this program.

/O   Overwrites <dfile> if it exists.

/L   Create MATLAB File (a binary file).

/M   Create MATRIXx File (an ASCII file).

/R   Output MATLAB or MATRIXx file matrices in a row instead of a column.

If neither /L nor /M option is specified, then an ASCII file is created.

Example 1

Convert your file to a PC-MATLAB file. Note that PC-MATLAB searches for a file with the suffix ''.MAT'' if no other suffix is specified. See appendix for more information on loading a file into PC-MATLAB.

> SYNTH63 FREQ C:FREQ.MAT /L

Example 2

Convert your file to MATRIXx row format:

> SYNTH63 FREQ A:\APPLIC\FREQ.MAT /M /R

Example 3

Output data to an ASCII file:

> SYNTH63 FREQ FREQ.ASC

Discussion

A synthesis table contains a maximum of 22 poles/zeros, poles/residues, or numerators/denominators (polynomial). It also contains the time delay, system gain, and either scale frequency (s-domain) or sample frequency and z power sign (z-domain).

The first 4 lines of the ASCII file contain the time delay, system gain, scale or sample frequency, and z power sign, respectively. The next line contains the # of poles (or numerators) followed by the # of zeros (or residues or denominators). Each of the following lines contain a complex pole (2 fields) followed by a complex zero (2 fields). This is repeated for all the poles and zeros in the synthesis table. If there are fewer zeros than poles, then the extra zeros contain the value 0.0, 0.0 (and vice versa). (MATLAB and MATRIXx files are not zero filled.)

Both the MATLAB and MATRIXx files contain the following matrix variables:

TimeDelay	Time delay (in seconds)
SysGain	System gain
SynthFreq	Scale frequency (Hz) if s-domain table Sample frequency (Hz) if z-domain table
PoleData	Vector of complex poles (if pole/zero or pole/residue table)
NumerData	Vector of complex numerators (if polynomial table)
ZeroData	Vector of complex zeros (if pole/zero table)
ResData	Vector of complex residues (if pole/residue table)
DenomData	Vector of complex denominators (if polynomial table)
Zpower	1 if negative z powers, 0 if positive z powers. (Only useful for z-domain table.)

X32TOBIN

Convert data from Intel Hex-32 source data file and put the results in the destination binary file.

Syntax   X32TOBIN <sfile> <dfile> [/U] [/O]

Options

<sfile>	Input Hex-32 data file.
<dfile>	Output binary file.
/U	Show help (usage information) for this program.
/O	Overwrite <dfile> if it exists.

Example

Convert an Intel Hex-32 file (FREQ.X32) to a binary data file (FREQ.BIN).

    X32TOBIN FREQ.X32 FREQ.BIN

Error Messages

Cannot open
output file

This error is generated if the output file cannot be opened with write permissions. Check for a valid path/file name. Also, check to see if the file is write protected.

Cannot open
input file

This error is generated if the input file cannot be opened with read permissions. Check for a valid path/file name. Also, check to see if the file is read protected.

Out of Memory

63TOSDF can generate this error if it cannot allocate enough memory to fully contain the data portion of the file specified. Check to see if the input file is a valid HP 3562A/3563A data file. If it is not, then garbage in, garbage out (or out of memory). If it is a valid file, then you need to free up some memory.

Unknown Option 'x'

This error may be generated by all the commands. Check your options. Typing the command with /U will list the valid options. Options are not case sensitive and not order dependant.

Viewdata

Introduction

The Viewdata utility allows you to display data graphically on your PC. You can display up to three traces of data from the same or different analyzers. For example, you can display a frequency response from an HP 3563A on trace A, a frequency response from an HP 35665A on trace B, and an FRF from an HP 3566A/3567A on trace C. You can also display a waterfall (or map) of results from an HP 35665A or HP 3566A/3567A.

The mouse can be used to move menus (left button to select item, right button to quit menu), and move the markers in the Trace Display mode (left button to zoom in, right button to zoom out).

Figure 5-1 is an example of the 3-trace display. In this example, each trace is a different scan (waterfall trace) from an HP 35665A waterfall.
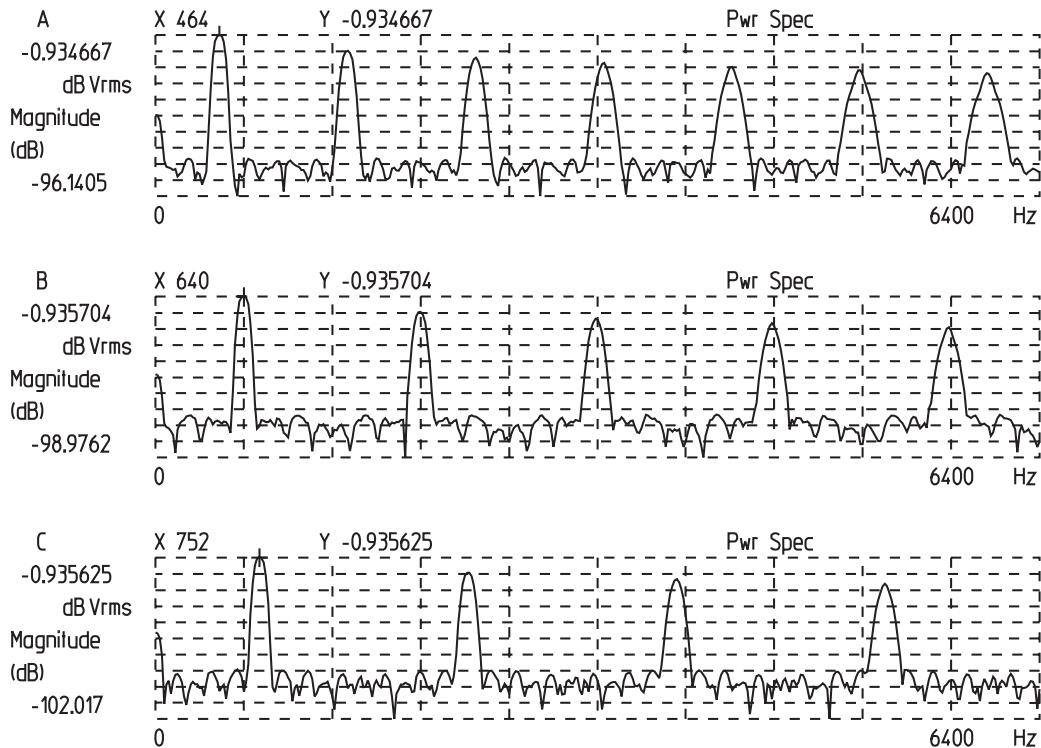
Figure 5-1. Three-trace display

Note

No file conversions are necessary for SDF or ASCII data. HP 3562A/3563A data must be converted from LIF to DOS format (see chapter 3), but need not be converted to SDF. If you set the file type to ''Auto,'' Viewdata will read data in any of these formats.

For data from any other supported analyzers, you must first convert the data to SDF format, then read it into Viewdata.

The utility includes marker, scale, zoom, and scroll functions. You can also plot or print the display.

Figure 5-2 shows a single-trace display. The annotation for each trace includes X and Y marker values, name of the measurement data (Pwr Spec in this example), the Y-axis coordinates, and the trace title (the date in this example).
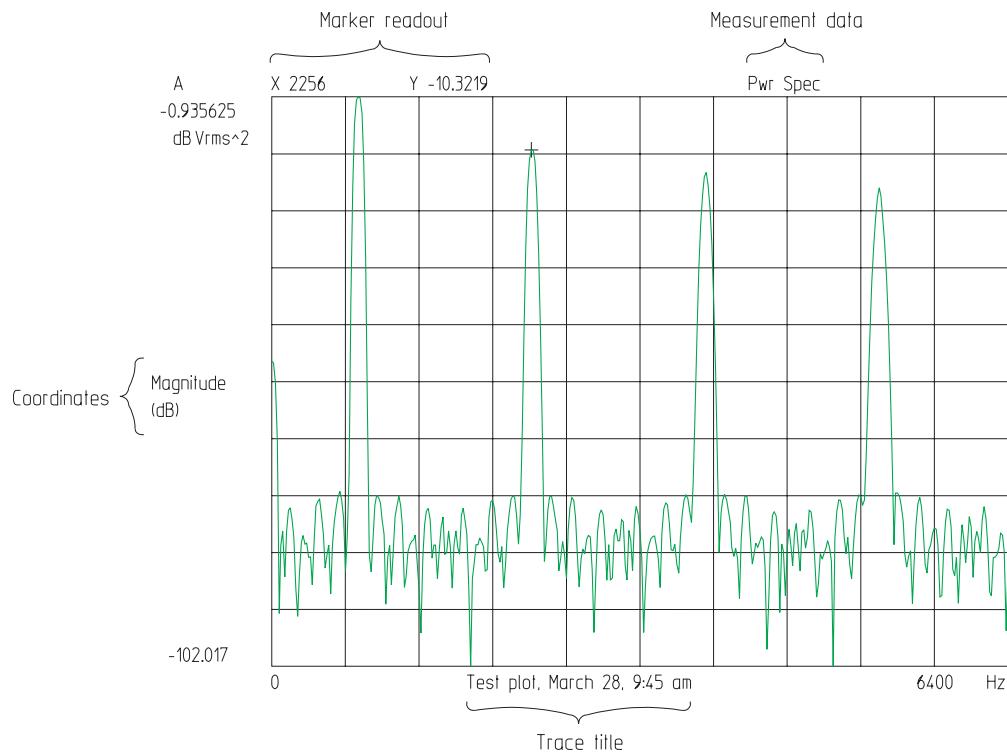
Figure 5-2. Single trace display

The menu line for Viewdata contains a menu name followed by a set of command words on the top line on the display. Each command has one letter in red on a color display (bright white on monochrome display) indicating the hot key for that command. The currently selected command is in inverse video (on color display, entire word in bright white on monochrome display).

There are two ways to execute a command:

- Press the hot key.

- Press the left or right arrow keys to change the active command, then press the [**Enter**] key to execute that command.

Figure 5-3 is the first screen that appears when you start Viewdata. The menu bar is across the top of the screen. In this example, File is the active command. The hot keys for the other commands are highlighted.

The rest of the screen contains information about the Viewdata setup and the data in the active trace. This information is updated whenever you change the setup or the active trace.


```
ViewData:  File  Display  Output  Save  Help  Quit

                        ─ DISPLAY TRACE A ─
     ACTIVE      Trace: A              Scan: 0
     FORMAT      Type: Auto
     TRACE       Data: Auto            Grid: Dots
                 AutoTitle:
     COORDS      Type: Auto
     X SCALE     Type: Auto
                 Spacing: Auto
     Y SCALE     Type: Auto
                 Units: Auto (RMS)     WfallHeight: 39.0%
                 dBmType: Auto         dBmRef: 50 Ohms
                 UserRef: 1
                        ─ DATA ─
                 Name:
                 Points: 0
                        ─ OUTPUT ─
                 Paging Printer on LPT1
```
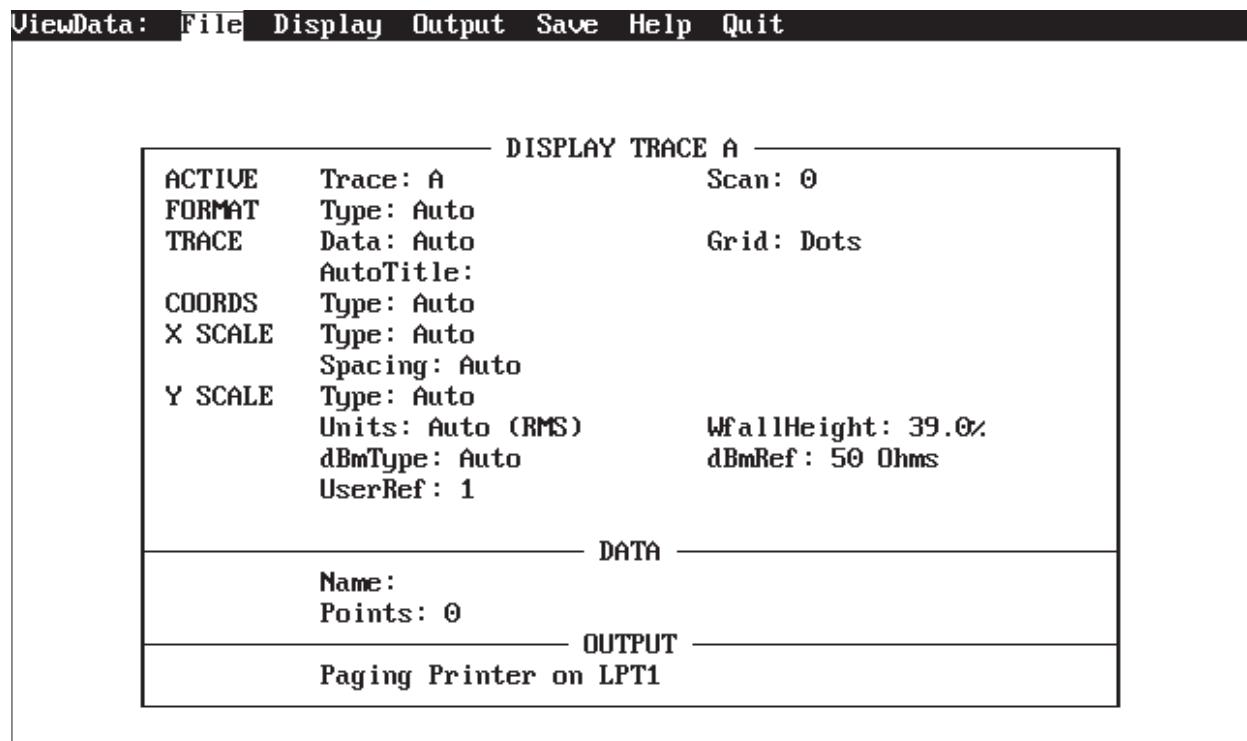
Figure 5-3. Viewdata main screen

When you change to the Output menu, the screen displays information on the output setup.
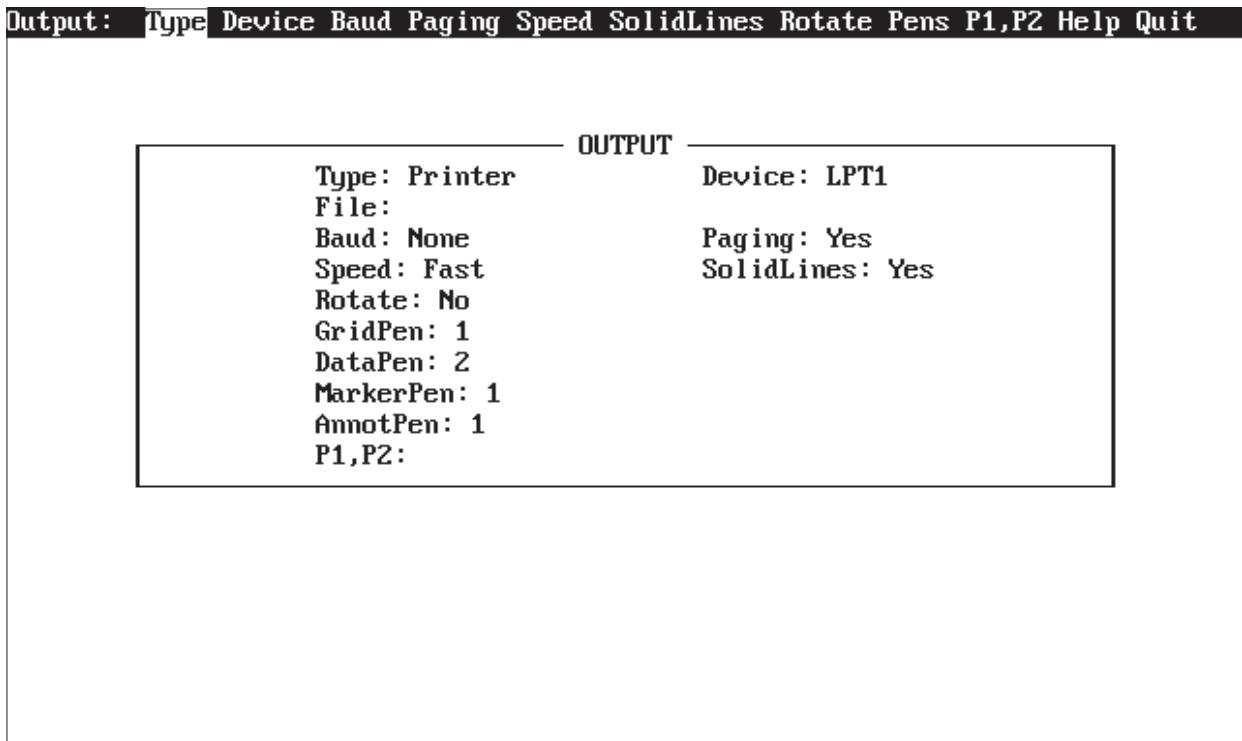
Figure 5-4. Output setup screen

VIEWDATA

Syntax

VIEWDATA [/D:<data>] [/S:<scans>] [/R:<row>] [<fileA>] [<fileB>] [<fileC>] [/U]
[/T:<title>] [/K:<keyFile>] [/E:<echoFile>] [/P] [/P:<plotfile>] [/I:<iniFile>]
[/A:<ascFile>] [/O:<ovlFile>] [/W:<wideCnt>] [/H:<highCnt>]
[/N:<pltNum>] [/M:<time>]

/D:<data>	Specify data result in file (default is result 0 if /P)
/R:<row>	Specify row in file (default is row 0 if /P)
/S:<scans>	Specify scan in file (default is scan 0 if /P). Can specify a range separated by ''-'' (for example, /S:0-5).
<fileA>	Data file to read into Trace A.
<fileB>	Data file to read into Trace B.
<fileC>	Data file to read into Trace C.
/U	Show help (usage information) for this command.
/T:<title>	Set the trace title (up to 30 characters) to be displayed on the X-axis of the data. If the title is empty, the measurement title from the data file is used. If you want to use a space in the trace title, surround the title with quotes (for example, "Trace Title").
/K:<keyFile>	Specify file from which to read key presses (redirect input).
/E:<echoFile>	Specify file to echo (record) key presses (useful for playing back keys using the /K option).
/P	Plot/print data and then exit program.
/P:<plotFile>	Plot/print data to specified file and then exit program. (See batch files ''60_PLOT.BAT'' and ''63_PLOT.BAT'' in appendix C.)
/A:<ascFile>	Dump ASCII data to specified file and then exit program.
/I:<iniFile>	Specify configuration file to read instead of VIEWDATA.INI. (See ''Config'' in the Save menu.)
/O:<ovlFile>	Specify file which contains HP-GL text/graphics to overlay upon the trace.
/W:<wideCnt>	The number of plots per page wide.

/H:\<highCnt\>	The number of plots per page high.
/N:\<pltNum\>	Which plot this is (relative to /W and /H). If not set when /W or /H used, then the plot number is automatically incremented & stored in 'VIEWDATA.INX' for the next plot.
/M:\<time\>	Display trace, wait \<time\> seconds, and exit program.

Example 1

To start Viewdata, type:

> VIEWDATA [**Enter**]

Example 2

To start Viewdata and display a particular file, type:

> VIEWDATA \<filename\> [**Enter**]

Example 3

To start Viewdata and display one file in trace A and another file in trace B, type:

> VIEWDATA \<filename1\> \<filename2\> [**Enter**]

Example 4

To start Viewdata and display one file in trace A, another file in trace B, and a third file in trace C, type:

> VIEWDATA \<filename1\> \<filename2\> \<filename3\> [**Enter**]

Example 5

To plot 3 traces side-by-side (wide) and 2 traces deep (high) for a group of files, type:

> REPEAT viewdata /w:3 /h:2 *.dat /p [**Enter**]

With the /W, /H, and /N options to VIEWDATA, you could specify where the current trace will be plotted on a page.

VIEWDATA menu

Viewdata:	**F**ile	**D**isplay	**O**utput	**S**ave	**H**elp	**Q**uit
Hot Keys:	F	D	O	S	H	Q

File   Set the file type and read data from a file.

Display   Set up the display parameters (format, coordinates, scaling).

Output   Set the output device (plotter/printer) characteristics.

Save   Save the program configuration to a file.

Help   Display help information.

Quit   Quit the Viewdata program.

File menu (if Type is Auto or HP 3563A/3562A)

File:	**T**ype	**R**ead	**H**elp	**Q**uit
Hot Keys:	T	R	H	Q

Type   Set the type of file (ASCII, SDF, or 3563/2 data file).

Read   Read a data file. It may contain any data. Neither Capture nor Throughput files
                    may be read directly (use DFDATA63 to extract data from these files).

Help   Display help information.

Quit   Quit the File menu and return to the Viewdata main menu.

Note   If Type is Auto and the name specified has a suffix of ''.BIN'' or ''.MUL'', the file is
                    assumed to be in HP 3569A BIN file format. 69TOSDF is invoked to convert the file to
                    a temporary SDF file and the SDF file is read and then deleted.

File menu (if Type is SDF)

File:	Type	Read	Header	Help	Quit
Hot Keys:	T	R	D	H	Q

Type   Set the type of file (ASCII, SDF, or HP 3563A/3562A data file).

Read   Read an SDF binary data file. ASCII SDF data files are not supported. An SDF file may contain multiple results of data where each result may have multiple rows, columns, or scans of data. After you specify the file name, the SDF data header is displayed allowing you to now specify the result, row, range of scans, and column (if there is more than one).

Header   Display the SDF data header for the specified file showing the results available to be read along with the number of rows and columns associated with each result.

Help   Display help information.

Quit   Quit the File menu and return to the Viewdata main menu.

File menu (if Type is ASCII)

File:	Type	Read	Points	Offset	Xvalues	Scans	Help	Quit
Hot Keys:	T	R	P	O	X	S	H	Q

Type   Set the type of file (ASCII, SDF, or 3563/2 data file).

Read   Read an ASCII file containing numbers. Numbers must be delimited by comma or
                     Viewdata will not read the imaginary data . (Use option "/B ," for SDFTOASC
                     when reading ASCII data into Viewdata.) Each line contains one point consisting
                     of an optional X value followed by a real Y value, followed by an optional
                     imaginary Y value.

Points   Specifies the maximum number of points in a scan. If there are more points in the
                     file, they are ignored.

Offset   Specifies a number of points at the beginning of the file to skip before reading the
                     specified points.

Xvalues   Specifies whether the first number on each line is an X value for the data point.

Scans   Specifies the maximum number of scans (or map lines) in the file. If there are more
                     scans in the file, they are ignored. The total number of values read from the file is
                     (Points * Scans).

Help   Display help information.

Quit   Quit the File menu and return to the Viewdata main menu.

Display menu

Display:	ShowTrace	Active	Format	Trace	Coords	Xscale	Yscale	Data	Help	Quit
Hot Keys:	S	A	F	T	C	X	Y	D	H	Q

ShowTrace   Display the data (with X marker).

Active   Specify A, B, or C as the active trace. The marker and zoom commands in the
                   ShowTrace menu affect only the active trace. For multi-scan data, you can also
                   specify which scan you want displayed in the trace.

Format   Specify that you want to display one, two, or three traces, or a waterfall on the
                   screen.

Trace   Set the type of trace drawing , the type of grid, and the type of trace title.

Coordinates   Set the Y-axis coordinates for the displayed data.

Xscale   Set auto, linear, or log X scaling for the displayed data.

Yscale   Set auto or fixed Y-axis scaling for the displayed data.

Data   Copy the trace to another file or remove the trace from the display.

Help   Display help information.

Quit   Quit the Display menu and return to the Viewdata main menu.

Active menu

File:	A	B	C	A&B	A/B/C	Scan	Help
Hot Keys:	A	B	C	&	/	S	H

A   Make trace A active.

B   Make trace B active.

C   Make trace C active.

A&B   Set the active Trace to the A & B traces. Any changes to the display attributes of the A trace are also changed for the B trace. In this mode, markers are moved on the A trace.

A/B/C   Set the active Trace to the A & B & C traces. Any changes to the display attributes of the A trace are also changed for the B & C traces. In this mode, markers are moved on the A trace.

Scan   Specify the active scan of a multi-scan trace (a waterfall or map). The first scan is scan 0. In the waterfall format, all scans are displayed, but the marker only applies to the active scan. In all other formats, only the active scan is displayed.

Help   Display help information.

Discussion

All trace commands (except Format) act only on the active trace. You can set different trace attributes for each trace. The marker moves only in the active trace.

To get data into a trace, either read data from a file or use CopyTo (under Display, Data) to copy data from one trace to another trace.

Format menu

File:	Auto	Single	Upper/Lower	Up/Mid/Low	Front/Back	Overlay	Waterfall	A_vs_B	Help
Hot Keys:	A	S	U	M	F	O	W	V	H

Auto
Determine the format from the data in the traces. If only one trace has data and it is the active trace, the format is single for single-scan data or waterfall for multi-scan data. If two traces have data, the format is upper/lower. If all three traces have data, the format is up/mid/low.

Single
Display one trace (the active trace).

Upper/Lower
Display two traces (A on top, B on the bottom).

Up/Mid/Low
Display three traces (A on top, B in the middle, and C on the bottom).

Front/Back
Display the A and B traces overlayed in the same grid area.

Overlay
Display the A, B, and C traces overlayed in the same grid area.

Waterfall
Display all scans of a multi-scan trace in a partially-overlapped form in a single grid area. To set the amount of overlap, use WfallHeight under Display, Yscale. To set the active scan, use Scan under Display, Active.

A_vs_B
Show Trace A Y-data vs. Trace B Y-data (on X-axis). Each trace's Y-data values are determined by the Coordinate and Yscale set in the respective active trace (e.g. can display Phase of Trace A vs. Imaginary part of Trace B). The number of display points is determined by Trace A. If Trace B has fewer points, then the last Y-data value is repeated. If Trace B has no data, then all of it's Y-data values are 0.0.

Help
Display help information.

Trace menu

Trace:	**D**ata	**G**rid	**T**itleType	**U**serTitle	**H**elp	**Q**uit
Hot Keys:	D	G	T	U	H	Q

Data   Set the data drawing to Auto, Points, Lines, Bars.

Grid   Sets the type of grid for the trace to None, Ticks, Dots, Lines, or Blank. If grids are turned on (Ticks, Dots, or Lines), then the type of grid that is drawn depends upon the coordinates. For LogMag, log spaced major & minor divisions are drawn. For Nyquist, the unit circle is drawn and horizontal/vertical lines crossing the origin. For all other coordinates, 10 equally spaced horizontal divisions are drawn. For log X, log spaced major & minor divisions are drawn. For linear X, 10 equally spaced vertical divisions are drawn.

TitleType   Select the information to be displayed below the trace.

UserTitle   Enter a trace title (up to 30 characters). This trace title is displayed if ''TitleType'' is ''User.'' (Entering a trace title automatically changes the TitleType to ''User''.)

Help   Display help information.

Quit   Quit the Display menu and return to the Viewdata main menu.

TitleType menu

Title Type:	Auto	User	Meas	ResBW	Date	Point
Hot Keys:	A	U	M	R	D	P

Auto | Use the other options from this menu in the following order:
|
| 1. If a user title has been defined, use it.
| 2. If there is a measurement title with the data, use it.
| 3. If the resolution bandwidth for the data is defined, use it.
| 4. If none of these are available, use no title.

User | Use the title specified under ''UserTitle.''

Meas | Use the SDF measurement title.

ResBW | Use the resolution bandwidth and sweep time from a swept spectrum measurement.

Date | Use the date from the measurement data.

Point | Use the point and direction from the SDF channel header.

TraceData menu

TraceData:	Auto	Points	Lines	Bars
Hot Keys:	A	P	L	B

Auto   Auto sets the type of drawing based upon the data. If the data is Octave data, the drawing is set to Bars. For other data the drawing is set to Lines.

Points   Points draws points only at the actual data points.

Lines   Lines draws connecting lines between data points.

Bars   Bars draws horizontal/vertical lines connecting data points (appears as bars).

Grid menu

Grid:	None	Ticks	Dots	Lines	Blank
Hot Keys:	N	T	D	L	B

None   Do not display a grid.

Ticks   Ticks only draws short marks at the edge of the grid (and no unit circle).

Dots   Dots draws dotted lines across the grid. (When you plot the display dotted lines
                        will plot as solid lines.)

Lines   Lines draws solid lines across the grid.

Blank   Turns off the square around the trace display and the active trace letter (e.g. 'A' in
                        upper left corner).

Coordinates menu

Coordinates:	Auto	Real	Imag	Mag	Log	dB	dBm	Inten	Phs	UnwrpPhs	Nyquist	Nichol	Help
Hot Keys:	A	R	I	M	L	D	B	T	P	U	Y	N	H

Auto — Set the coordinate based upon the data. If the data type is intensity, then Intensity is chosen. If the data is frequency domain (X-axis units of Hz, RPM, or Orders), then dB is chosen. If the data is orbit data, then Nyquist is chosen. If the data is complex, then Mag is chosen. Otherwise, Real is chosen.

Real — Show the real part of the data.

Imag — Show the imaginary part of the data. If the data is not complex then show zero.

Mag — Show the magnitude of the data (always positive).

Log — Show the magnitude of the data on a log Y-axis.

dB — Show the magnitude of the data in deciBels (10 * log (real^2 + imag^2)).

dBm — Show the magnitude of the data in dBm units (power referenced to 1 mWatt). To set reference impedance, press:

Yscale, Units, dBmReference or dBmType

Inten — Show Intensity, which is the Magnitude of the data in dB units, with the sign negated if the original real part of the data is negative. If data is set to bars, then values that have negative real parts are shown with solid bars.

Phs — Show the wrapped phase of data. The wrapped phase is shifted to between –180 and +180 degrees. If the data is not complex, the phase shown is 0.

UnwrpPhs — Show the unwrapped phase of the data. This is the actual phase referenced to the first data point.

Nyquist — Show Imaginary vs. Real.

Nichol — Show dB vs. Phase.

Help — Display help information.

Discussion

In both dB and dBm, you can specify a magnitude reference value. To set the user reference, press:

Yscale, Units, UserReference

Xscale menu

Xscale:	Type	Left	Right	Spacing	Help	Quit
Hot Keys:	T	L	R	S	H	Q

Type   Set auto or fixed scale for the X-axis.

Left   Specify an X limit for the left side of the screen (only if the scale type is fixed).

Right   Specify an X limit for the right side of the screen (only if the scale type is fixed).

Spacing   Set auto, linear, or logarithmic X-axis spacing.

Help   Display help information.

Quit   Quit the Xscale menu and return to the Display menu.

Xscale Spacing menu

Xscale:	Auto	Linear	Log
Hot Keys:	A	L	G

Auto   Set X-axis spacing to be the same as the resolution of the data. If the data resolution is arbitrary, then linear X-axis is used.

Linear   Use a linear X-axis.

Log   Use a logarithmic X-axis.

Xscale Type menu

Xscale:	Auto	Fixed
Hot Keys:	A	F

Auto   Show all of the data. Whenever new data is read, the X scale is set to auto.

Fixed   Show only the data between the specified Left and Right X values.

Yscale menu

Yscale:	Type	To**p**	**B**ottom	Units	**W**fallHeight	**H**elp	**Q**uit
Hot Keys:	T	P	B	U	W	H	Q

Type   Set auto or fixed Y-axis scaling for the displayed data.

Top   Specify a Y-axis limit for the top part of the screen.

Bottom   Specify a Y-axis limit for the bottom part of the screen.

Units   Specify the Y-axis units for frequency domain data.

WfallHeight   Specify how much the traces in the waterfall display should overlap (in percent of total trace height). If you specify 0%, the traces do not overlap at all. If you specify 100%, the traces overlap completely (similar to a ''front/back'' format).

Help   Display Help information.

Quit   Quit the Yscale menu and return to the Display menu.

Yscale Type menu

Yscale:	**A**uto	**F**ixed
Hot Keys:	A	F

Auto   Show all of the data. Whenever new data is read, the Y-axis scale is set to auto.

Fixed   Show only the data between the specified Top and Bottom Y-axis values.

Units menu

Units:	Peak/RMS	Power	Density	dBmType	dBmReference	UserReference	Quit
Hot Keys:	P	W	D	T	R	U	Q

Peak/RMS   Specify whether the data will be displayed in peak or rms units. If you choose Auto, the units will be the same as the data. SDF files are always peak. HP 3563A/62A data may be peak or rms.

Power   Specify the power of the Y-axis units (volts or volts^2). If you choose Auto, the displayed unit power is the same as the data unit power.

Density   Specifies whether the data will be displayed in density units (/Hz). If Auto is chosen, then the data will ONLY be shown in density units for PSD data (i.e. not Power Spectrums). If /Hz is chosen, then spectrum data will be shown in V/rtHz or V^2/Hz depending on the ''Power'' selection.

dBmType   Specify whether the reference impedance used in dBm coordinates is taken from the data file input impedance (Auto) or the dBmReference is used (Fixed). If the input impedance from the data file is >= 1 MOhm, then it is ignored and the dBmReference is used.

dBmReference   The reference value used for dBm coordinates (see dBmType for more details).

UserReference   A user reference value used for dBm or dB coordinates. It is a magnitude value, so an entered value of 1.0E-6 would represent $20 * \log10 (1.0E\text{-}6) = -120$ dB reference (i.e. add 120 dB to the trace values).

Data menu

Data:	CopyTo	Remove	Help	Quit
Hot Keys:	C	R	H	Q

CopyTo   Copy the active trace to either of the 2 other traces. All trace display attributes
                 associated with the trace are copied to the destination trace with the data. Any data
                 that was previously in the destination trace is removed.

Remove   Remove all data associated with the specified trace. It does not affect any display
                 attributes associated with the specified trace.

Help   Display help information.

Quit   Quit the Data menu and return to the Display menu.

ShowTrace menu

Display:	Xpos	Peak	Min	⌊	Mkr	™	⌈	Zoom	Σ
Hot Keys:	X	P	M	⌊		™	⌈		Σ

Display:	Home	<PgUp	Move	PgDn>	End	+	Menu	–	Ins	Mkr	Del	Output	Quit
Hot Keys:	Home	PgUp		PgDn	End	+		–	Ins		Del	O	Q

A   Make trace A active.

B   Make trace B active.

C   Make trace C active.

S   Specify the active scan.

Xpos   Enter a new X-axis value for the marker. The marker moves to that value in the active trace.

Peak   Move the marker to the largest Y-axis value in the trace.

Min   Move the marker to the smallest Y-axis value in the trace.

⌊   Move the marker one point to the left. If you hold the key down, the marker keeps moving to the left.

™   Move the marker one point to the right. If you hold the key down, the marker keeps moving to the right.

⌈   Click the left mouse button or press the up arrow key on the keyboard to Zoom in by factor of 2, centered about marker. (Display half as many points as the current display.)

Σ   Click the right mouse button or press the down arrow key on the keyboard to Zoom out by factor of 2, centered about marker. (Display twice as many points as the current display.)

Home	Move to beginning of the data when the display is zoomed.
PgUp	Scroll half a screen towards beginning of the data when the display is zoomed.
PgDn	Scroll half a screen towards end of the data when the display is zoomed.
End	Move to end of the data when the display is zoomed.
+	Turn the menu on.
-	Turn the menu off. This command is useful if you want to print the screen (using the DOS Print Screen function) without extraneous information.
Ins	Turn the marker on.
Del	Turn the marker off.
/	Turn reference marker on (and move it to current marker position)
?	Turn reference marker off.
Output	Output display directly to plotter/printer. To set up for plotting/printing, use the output menu under the Viewdata main menu.
D	Dump trace data in ASCII to a file.
T	User entered Text mode.
G	User entered Graphics mode.
W	Write overlayed text/graphics to a file.
R	Read overlayed text/graphics from a file.
F2	Delete all overlayed text/graphics (created w/ 'T' or 'G').
Quit or Esc	Quit the Showtrace menu and return to the Display menu.
F8	Quit Viewdata.

User Entered Graphics

With the 'G' command in the trace display mode, graphic lines and graphic (vector) text may be overlayed anywhere upon the trace display. This allows the user to point to some region of interest (using a simple line with an arrow) and entering text of any size and orientation (45 degree increments).

In addition, any HP-GL file may be imported and displayed anywhere on the trace display. This allows the user to display a company logo or show a graphic that explains how or where the measurement was made. HP-GL files may be generated with DRAWING GALLERY (plot to a file). The arrow keys or the mouse move a graphic marker over the trace display. To draw a line, a reference marker must be placed upon the display. To do this, move the marker to where you want the line to start and either press Insert or the Left mouse button (a small square will be displayed). This is creates a reference marker. Now move the marker to where you want the end of the line and press Enter or the Right mouse button (now the line is drawn). To do another line repeat the previous steps. To draw a sequence of connected lines, you may leave out the press of Enter (or Right button) as starting a new line finishes drawing an old line.

To import a graphic, place the reference marker (by press an Insert key or click the left mouse button) at one edge of the location to put the graphic, move the marker to the opposite edge to put the graphic and then press 'I'. The following commands are recognized in graphics mode:

↓ ™ ⌈ ∑	(arrow keys or use mouse) move marker,
Ins	Set Reference Marker. If Reference Marker already set, then this draws a line between marker & reference marker,
L Mouse Button	Same as 'Ins',
Enter	Draw line between marker & ref marker (and turns off ref marker),
+	Add an arrow at end of nearest line,
-	Dashed line on/off,
Del	Delete nearest graphics line/text,

R Mouse Button   Same as 'Del',

I   Import text/graphics from a file. The information is scaled to fit between the marker & reference marker,

T   Enter text at marker position,

A   Set angle of entered text (multiples of 45 deg),

S   Set size (height/width) of entered text (where 1.0 is normal size),

Home   Top of page,

End   Bottom of page,

F2   Delete all graphics,

Esc   Quit graphics mode and return to the ShowTrace menu.

User Entered Text

With the 'T' command in the trace display mode, text may be overlayed anywhere upon the trace display. This allows the user to enter information anywhere within the text grid area, and also the ability to overwrite any existing annotation. You can overwrite existing annotation by moving the cursor to the annotation (using the arrow keys or the mouse) and typing text over the annotation or erase the annotation by typing spaces. The following commands are recognized in text mode:

Any printable char	Overwrite character at cursor,
↓ ™ ⌠ ∑	(arrow keys or use mouse) move cursor,
Ins	Toggle Insert character mode. This allows text to be inserted before the character at the cursor position.
L Mouse Button	Same as 'Ins',
Del	Delete character at cursor,
Backspace	Delete character before cursor,
R Mouse Button	Same as 'Del',
Home	Top of page,
End	Bottom of page,
F2	Delete all text,
Esc	Quit text mode and return to the ShowTrace menu.

Output menu

Output:	Type	Device	Baud	Paging	Speed	SolidLines	Rotate	Pens	P1,P2	Help	Quit
Hot Keys:	T	D	B	G	S	L	R	P	1	H	Q

When the trace is displayed, pressing the ''O'' key will output the screen to the selected printer or plotter. This menu selects the parameters controlling output to the plot/print device.

Type

Set the type of output device (printer, plotter, or PrintScreen printer). Only HP-GL plotters are supported for ''plotter,'' and PCL printers (for example, DeskJet, LaserJet) are supported for ''printer.'' Either the ''PrintScreen'' device type or the DOS PrintScreen key may also be used to generate direct printer dumps. In this case, the DOS ''GRAPHICS'' command must be used to select the printer type (for example, GRAPHICS DESKJET).

Device

Set the destination device of the output operation. Plot/print operations are supported to the parallel printer ports (LPT1 through LPT4), serial ports (COM1 through COM4), or to a file. Serial plotters/printers must be set up as 8 bit data, 1 stop bit, no parity. Serial printers need to be set up to allow XON/XOFF handshaking. Serial plotters use software handshaking. (This setting does not affect the output from the DOS PrintScreen function.)

Baud

Set the baud rate of the serial plotter/printer (NONE, 1200, 2400, 4800, or 9600 baud). If NONE is specified, then the serial port is not configured by VIEWDATA. In this case, the serial port must be set up with the DOS ''MODE'' command (for example MODE COM1:96,N,8,1,B).

Paging

Specify whether a ''PG'' command will be sent at the end of a plot (useful for paging plotters), or a form feed will be sent at the end of a print.

Speed

Set the speed of the plotting operation using the ''VS'' command. ''Fast'' sets the speed to the default for the plotter (50 cm/s with fiber tip pens) and ''Slow'' sets the speed to 10 cm/s.

SolidLines

Specify whether dotted lines (in the grid) will be forced to be solid lines when plotting.

Rotate   Specify whether the plot or print output will be rotated 90 degrees. On a plotter, this uses the ''RO'' command. This setting does not affect the DOS PrintScreen function.

Pens   Set the plotter pen number to be used in plotting the following information:
    Grid
    Data trace
    Marker
    All other Annotation.

A pen number of zero will cause the selected field to be NOT plotted.

P1,P2   Set the P1,P2 plotter limits. The actual numbers entered are P1x,P1y,P2x,P2y. See your plotter manual for a description of the ''Input P1 and P2 Instruction (IP)'' and the parameter limits for your particular plotter.

Help   Display help information.

Quit   Quit the Output menu and return to the Viewdata main menu.

Save menu

Save:	Config	Quit
Hot Keys:	C	Q

Config   Save the current configuration to the file ''VIEWDATA.INI'' in the same directory
                  as ''VIEWDATA.EXE''.

                  If you want to make a permanent copy of a configuration file, use the DOS copy
                  command to copy VIEWDATA.INI to another file name. Then when you start
                  Viewdata, you can specify that file name with the /I: option.

                  For example, copy VIEWDATA.INI to FRF.INI. Then start Viewdata by typing:

                  VIEWDATA /I:FRF.INI

                  Viewdata will come up with the configuration from the file FRF.INI rather then
                  VIEWDATA.INI.

Quit   Quit the Save menu and return to the Viewdata main menu.

6

Using SDF Data with C Programs

SDF Library Functions

This chapter provides information on the SDF library functions. These functions are located in \SDFLIB\SDFUTIL.LIB. The SDF library functions are written on a much higher level to help you more easily get at the data that you need.

When using these functions, include these include files (located in \SDFLIB) in your C programs:

- SDF_COMM.H
- SDF_BASI.H
- SDF_UTIL.H

There are three C example programs (located in \SDFLIB) that show you how to use the SDF Library functions:

SDF.C   Prints the headers of an SDF file.

SDFDATA.C   Prints all the data (applying corrections) in an SDF file.

SDFMPY.C   Multiplies first data result in the SDF file by the factor specified. This program demonstrates reading and writing data to an SDF file.

Caution   SDFMPY.C changes data in the selected SDF file. Make a backup copy of your SDF file if you want to preserve its contents.

The Standard Data Format

All measurement data stored by the HP 3566A, HP 3567A, HP 35665A, HP 35670A, HP 89410A, and HP 89440A is in a file format called Standard Data Format (SDF). Data saved in Standard Data Format is the pure measurement data, with no engineering units or window corrections applied to it. It is stored in binary, along with accompanying header information, and represents the highest resolution data available from the analyzer. An SDF file is composed of several headers followed by the measurement data. The headers contain information such as measurement settings that help reconstruct the data.

Appendix B, ''Standard Data Format,'' describes the structure of SDF in detail.   Read appendix B to become familiar with SDF.

SdfOpen

Syntax   short SdfOpen (char far *filename);

Returns   an integer(short) between 0 and 5 containing a file number. A return of 0 indicates
an error; 1 to 5 indicates a valid file.

This function attempts to open an SDF file with the indicated file name. If the file is found, the format
type of the file is read. If it is a binary SDF file, the file number is recorded in the current list of open
SDF files and returned. The returned file number may then be passed to other SDF functions to specify
the file.

SdfClose

Syntax   int SdfClose (short fn);

Returns   an integer(short) of zero or minus one. A return of 0 indicates no error; a return of
-1 indicates an error.

This function closes an open file with the indicated file number and decrements the current list of open
SDF files.

SdfFileHdr

Syntax   int SdfFileHdr (short fn, struct SDF_FILE_HDR far *fileHdr);

Returns   an integer(short) of zero or minus one. A return of 0 indicates no error; a return of -1 indicates an error.

This function reads the file-header information from an open SDF file, where *fn* is the number of the open SDF file (for details, see SdfOpen). The information is put in the SDF_FILE_HDR structure, where fileHdr is the pointer to the structure.

The SDF file header record contains information such as the application (instrument) that created it, the time stamp, the index of all other types of records in the file and the offsets of those records from the beginning of the file.

SdfMeasHdr

Syntax   int SdfMeasHdr (short fn, struct SDF_MEAS_HDR far *measHdr);

Returns   an integer(short) of zero or minus one. A return of 0 indicates no error; a return of -1 indicates an error.

This function reads the measurement-header information from an open SDF file, where *fn* is the number of the open SDF file (for details, see SdfOpen). The information is put in the SDF_MEAS_HDR structure, where measHdr is the pointer to the structure.

The SDF measurement header record contains information such as the frequency span, start and center, block size, average type and count, etc. There is always one measurement header record in every SDF file.

SdfDataHdr

Syntax   int SdfDataHdr (short fn, short dataNo ,
                    struct SDF_DATA_HDR   far   *dataHdr);

Returns   an integer(short) of zero or minus one. A return of 0 indicates no error; a return of
                     -1 indicates an error.

This function reads the data-header information from an open SDF file, where *fn* is the number of the
open SDF file (for details, see SdfOpen). The information is put in the SDF_DATA_HDR structure,
where dataHdr is the pointer to the structure.

Note   The *dataNo* parameter selects the data result that you want to examine. Most
                  instruments have only 1 data result, therefore, you will usually specify 0 (zero) for this
                  parameter, which selects the first data result.

The SDF data header record contains information for a specific measurement data type. It contains the
index to the first vector header record as well as the total number of rows and columns (vectors)
associated with it. It also contains fields describing the data type, number of points per trace, X-axis
spacing information, Y-axis data format. There is one data header record for every measurement result
type in the SDF file.

SdfVectHdr

Syntax   int SdfVectHdr (short fn, short dataNo, short rowNo,
                    short colNo, struct SDF_VECTOR_HDR   far   *vectHdr);

Returns   an integer(short) of zero or minus one. A return of 0 indicates no error; a return of
                    -1 indicates an error.

This function reads the vector-header information from an open SDF file, where *fn* is the number of the
open SDF file (for details, see SdfOpen). The information is put in the SDF_VECTOR_HDR structure,
where vectHdr is the pointer to the structure.

Note   See SdfYdata for a description of the parameters used with this function.

Two fields in the vector header record, the_CHANNEL_record[0] and the_CHANNEL_record[1],
contain indexes to channel header records containing the channel numbers associated with that vector.
Although the the_CHANNEL_record[0] and the_CHANNEL_record[1] fields do not refer to actual
channel numbers themselves, in the HP 3566A/3567A the channel number is always 1 plus the value in
these fields, since channel header records are a contiguous block starting from channel 1 and the indexes
to the channel header records start at 0. For single channel measurements, the_CHANNEL_record[1]
will always be –1.

The vector header record is mainly used to store the indexes of associated channel header records and the
exponents (power) applied to the corresponding trace data.

SdfChanHdr

Syntax   int SdfChanHdr (short fn, short chanNo,
               struct SDF_CHANNEL_HDR   far   *chanHdr);

Returns   an integer(short) of zero or minus one. A return of 0 indicates no error; a return of
              -1 indicates an error.

This function reads the channel-header information from an open SDF file, where *fn* is the number of the open SDF file (for details, see SdfOpen). The information is put in the SDF_CHANNEL_HDR structure, where chanHdr is the pointer to the structure.

There is a separate channel header for each channel. Use *chanNo* to select a channel.

Note   A *chanNo* of zero selects channel 1; a *chanNo* of one selects channel 2, and so forth.

The SDF channel header record contains information pertaining to the channel from which the data was received as well as the window function and engineering unit correction factors to be applied to the data.

SdfScanStruct

Syntax   int   SdfScanStruct (short fn,
                      struct SDF_SCAN_STRUCT far *scanStruct);

Returns   an integer(short) of zero or minus one. A return of 0 indicates no error; a return of
                      -1 indicates an error.

This function reads scan information from an open SDF file, where *fn* is the number of the open SDF file
(for details, see SdfOpen).   The information is put in the SDF_SCAN_STRUCT structure, where
scanStruct is the pointer to the structure.

SdfScanBig

Syntax   int SdfScanBig (short fn, struct SDF_SCAN_BIG far *scanBig);

Returns   an integer (short) of greater than zero, zero, or minus one. A return of greater than zero indicates that an SDF_SCAN_BIG header was found (returns SDF_SCAN_BIG_TYPE) or one was created from an SDF_SCAN_STRUCT (returns SDF_SCAN_STRUCT_TYPE). A return of 0 indicates that no scan header of any type was found. A return of −1 indicates an error.

This function reads the scan information from an open SDF file, where *fn* is the number of the open SDF file (for details, see SdfOpen). The information is put in the SDF_SCAN_BIG structure, where *scanBig* is the pointer to the structure. If an SDF_SCAN_BIG record is not in the file, one will be created from the SDF_SCAN_STRUCT record in the file.

SdfScanVar

Syntax   int SdfScanVar (short fn, short varInx, struct SDF_SCAN_VAR far *scanVar);

Returns   an integer (short) of zero or minus one. A return of 0 indicates no error; a return of −1 indicates an error.

This function reads the scan variable information from an open SDF file, where *fn* is the number of the open SDF file (for details, see SdfOpen). The information is put in the SDF_SCAN_VAR structure, where *scanVar* is the pointer to the structure.

The *varInx* parameter selects the scan variable that you want to examine. The first variable (*varInx* equal to zero) is actually an SDF_SCAN_VAR record created from the SDF_SCAN_STRUCT record in the SDF file. Most instruments don't have true SDF_SCAN_VAR records, but do have the SDF_SCAN_STRUCT record, so you will usually specify 0 (zero) for this parameter. The HP 89410A and HP 89440A do use 3 addition scan variable records for waterfall data, so you may specify a *varInx* from 0 to 3.

SdfScansUsed

Syntax   long SdfScansUsed (short fn);

Returns   A long integer of greater than or equal to minus one. The return value is the
number of scans in the SDF file. A return of 0 indicates no scan data; a return of –1
indicates an error.

This function returns the number of scans in an open SDF file, where *fn* is the number of the open SDF
file (for details, see SdfOpen).

The number of scans is in the SDF_SCAN_BIG header if it exists in the file, or the
SDF_SCAN_STRUCT header if it exists in the file. If neither of these headers are in the file, then there
is no scan data.

SdfYdata

Syntax   int SdfYdata (short fn, long offset, short dataNo,
short rowNo, short colNo, long scanNo, long pnts,
short dataType, float   far   *y);

Returns   an integer(short) of zero or minus one. A return of 0 indicates no error; a return of
-1 indicates an error.

This function reads Y-axis data from an open SDF file and puts it in an array, where *y* is the pointer to the
array, and *fn* is the number of the open SDF file (for details, see SdfOpen).

The *offset* is the number of data points to skip before reading the data. One situation where an offset is
useful is if you are reading data in multiple passes. If you don't want to use an offset, specify an offset of
zero.

The *dataNo* selects the data result that you want to examine–most instruments have only 1 data result,
therefore, you will usually specify zero for this parameter, which selects the first data result. In general,
rowNo selects the channel, scanNo selects the trace (in the case of waterfall or spectrogram
measurements), and colNo is not used.   A rowNo of zero selects channel 1 and a scanNo of zero selects
the first trace. For further details, see chapter 2.

Note   All parameters are zero based. Therefore, for parameters such as dataNo, rowNo,
colNo, and scanNo, zero selects the first item, as described in the above paragraph.

Furthermore, you must include all parameters. For unused parameters,
specify zero.

The *pnts* are the number of data points you want to read. For example, a complex value is 2 points.   To
determine the number of data points in the record, see ''Reconstructing a Trace'' in appendix B.

The *dataType* is the size of the data block (the format of the data in memory), as shown in the following table.

Available Datatypes

DataType	Conversion performed
0 (zero)	no conversion, data is same format as in the file
SHORT16	convert to 16-bit integer
LONG32	convert to 32-bit integer
FLOAT32	convert to 32-bit floating point
DOUBLE64	convert to 64-bit floating point

For example, suppose the data in your file is in integer format and you use SdfYdata with a dataType of FLOAT32. In this case, SdfYdata converts the data from your file into 32-bit floating point and scales the data based on the value of channelScale (in SDF_CHANNEL_HDR).

Note   When the dataType specified is floating point (FLOAT32 or DOUBLE64), Y-axis data is read with window-correction and engineering-unit factors applied, and it is written back to the SDF file with these factors removed.

SdfYdataWrite

Syntax   int SdfYdataWrite (short fn, long offset, short dataNo,
                    short rowNo, short colNo, long scanNo, long pnts,
                    short dataType, float   far   *y);

Returns   an integer(short) of zero or minus one. A return of 0 indicates no error; a return of
                    -1 indicates an error.

This function writesY-axis data from an array and puts it in an open SDF file , where y is the pointer to
the array, and *fn* is the number of the open SDF file (for details, see SdfOpen).

When you use SdfYdataWrite, data is converted back into its original form before writing to your file.
For example, suppose the data in your file is in integer format and you use SdfYdata with a dataType of
FLOAT32. In this case, SdfYdata converts the data from your file into 32-bit floating point and scales the
data based on the value of channelScale. If you use SdfYdataWrite to write data back to your file, the
data is converted back into integer form before it is written to your file. If any data value exceeds
channelScale (in SDF_CHANNEL_HDR), that data will be clipped.

Note   When the dataType specified is floating point (FLOAT32 or DOUBLE64), Y-axis data
                    is read with window-correction and engineering-unit factors applied, and it is written
                    back to the SDF file with these factors removed. For further details about dataType, and
                    for a description of the parameters used with this function, see SdfYdata .

SdfXdata

Syntax   int SdfXdata (short fn, long offset, short dataNo,
                     short rowNo, short colNo, long pnts, short dataType,
                     float   far   *x);

Returns   an integer(short) of zero or minus one. A return of 0 indicates no error; a return of
                     -1 indicates an error.

This function reads X-axis data from an open SDF file and puts it in an array, where x is the pointer to the array, and *fn* is the number of the open SDF file (for details, see SdfOpen).

An xdata header is only included in the SDF file for data types that use arbitrary X-axis spacing, otherwise all X-axis spacing is defined by fields in the data record header.

The xdata header contains all of the X-axis data for a single vector and corresponds to the equivalent ydata header.

Note   See SdfYdata for a list of valid dataTypes and for a description of the parameters used
                     with this function.

SdfScanData

Syntax   int SdfScanData (short fn, long offset, short varInx,
                  long pnts, short dataType, float   far   *s);

Returns   an integer(short) of zero or minus one. A return of 0 indicates no error; a return of
                  -1 indicates an error.

This function reads scan data from an open SDF file and puts it in an array, where *s* is the pointer to the
array, and *fn* is the number of the open SDF file (for details, see SdfOpen).

In the HP 3566A, 3567A and 35665A, map lines are stored as scan data blocks.

The *varInx* parameter selects the scan variable data that you want to read. The first variable (*varInx* equal
to zero) is actually scan data from the SDF_SCAN_STRUCT record in the SDF file. Most instruments
don't have SDF_SCAN_VAR records, but do have the SDF_SCAN_STRUCT RECORD, so you will
usually specify 0 (zero) for this parameter. The HP 89410A and HP 89440A do use 3 addition scan
variable records for waterfall data, so you may specify a *varInx* from 0 to 3.

SdfTrunc

Syntax   long SdfTrunc (short fn);

Returns   a long integer or minus one (-1 indicates an error).

This function returns a long integer. This integer is the accumulated number of times that data was truncated by any of these functions:

- SdfYdata()
- SdfYdataWrite()
- SdfXdata()
- SdfScanData()

Separate truncation counts are kept for each open file. Therefore, you must specify the number (fn) of the open SDF file with this function (for details about *fn*, see SdfOpen).

The count for an SDF file is cleared (reset) when the file is open or after calling this routine.

SdfCommentHdr

Syntax   int SdfCommentHdr (short fn, short commNo, struct
            SDF_COMMENT_HDR far *commHdr);

Returns   an integer (short) of zero or minus one. A return of 0 indicates no error; a return of
            −1 indicates an error.

This function read the comment-header information from an open SDF file, where *fn* is the number of the open SDF file (for details, see SdfOpen). The information is put in the SDF_COMMENT_HDR structure, where *commHdr* is the pointer to the structure.

The *commNo* parameter selects the comment header that you want to examine. Most instruments have only one comment header, so you will usually specify 0 (zero), which selects the first comment.

The SDF comment header record contains information about a comment associated with this data file. It contains the type of comment, the length of the comment, the maximum length of the comment, and to what part of the file the comment applies. The actual comment follows the comment header. Use the SdfCommentRead function to read the comment, and the SdfCommentWrite function to write to the comment area.

SdfCommentRead

Syntax   long SdfCommentRead (short fn, short commNo, long offset, long size, char far *buf);

Returns   a long integer of greater than or equal to minus one. The return value is the number of characters of comment actually read; a return of –1 indicates an error.

This function reads comment data from an open SDF file and puts it in a character array, where *buf* is the pointer to the array, and *fn* is the number of the open SDF file (for details, see SdfOpen).

The *commNo* parameter selects the comment header from which you want to read. Most instruments have only one comment header, so you will usually specify 0 (zero), which selects the first comment.

The *offset* parameter is the number of characters to skip before reading the data. One situation where an offset is useful is if you are reading data in multiple passes. If you don't want to use an offset, specify an offset of zero.

The *size* parameter is the maximum number of characters you want to read. This function returns the actual number of characters that you read.

SdfCommentWrite

Syntax   long SdfCommentWrite (short fn, short commNo, long offset, long size, char far *buf);

Returns   a long integer of greater than or equal to minus one. The return value is the number of characters of comment actually written; a return of −1 indicates an error.

This function writes a character array to the comment area in an open SDF file, where *buf* is the pointer to the array, and *fn* is the number of the open SDF file (for details, see SdfOpen).

The *commNo* parameter selects the comment header to which you want to write. Most instruments have only one comment header, so you will usually specify 0 (zero), which selects the first comment.

The *offset* parameter is the number of characters to skip before writing the data. One situation where an offset is useful is if you are writing data in multiple passes. If you don't want to use an offset, specify an offset of zero.

The *size* parameter is the maximum number of characters you want to write. This function returns the actual number of characters that you wrote. Since the comment area is a fixed size (fixed at the time of file creation) you may not be able to write the number of characters specified.

This function automatically updates the comment size in the appropriate SDF_COMMENT_HDR.

MATLAB MEX-Files & M-Files

Overview

MATLAB lets you call user programs (MEX-files) from MATLAB as if they were built-in MATLAB functions. MEX-files are usable only on PC-MATLAB or AT-MATLAB. They do not work on 386-MATLAB.

Note   MATLAB, PC-MATLAB, and AT-MATLAB are trademarks of
The MathWorks, Inc.

M-files are command scripts. MATLAB executes the commands found in M-files, instead of getting input from the keyboard.

This section describes the MEX-files and M-files for MATLAB that operate on SDF data.

Installing MEX-Files and M-files

To install the MEX-files and the M-files, copy *.m and *.mex (in the \matlab directory) to any directory on your MATLAB search path (see MATLABPATH in your MATLAB User's Guide). Copy matdemo.tim to this directory: \analyzer\tc. Copy sdftoml.exe to this directory: \analyzer\util (sdftoml.exe is available as part of the Standard Data Format Utilities).

filtrsdf.mex

Syntax   filtersdf (b, a, name, dataNo, rowNo, colNo)

Returns   not applicable

This MATLAB subroutine digitally filters SDF data (time capture data).

Caution   This MEX-file will modify your SDF file. You may want to make a back up copy of the
✋   SDF file before using this subroutine.

Parameter Definitions

Parameter	Definition
b	Digital filter coefficients (numerator)
a	Digital filter coefficients (denominator)
name	name of SDF file to filter
dataNo	data result to filter (default = 0, first result)
rowNo	row number to filter (default = 0, first row)
colNo	column number to filter (default = 0, first column)

All scans in a file are filtered. If the result is a time capture, the filter is cleared only once at the start of
the data. Otherwise, the filter is cleared at the start of each scan.

For a time capture file, dataNo is 0 (captured data is always the first result in the file), rowNo is the
channel number (0 is channel 1, 1 is channel 2), and colNo is always 0. For example, to filter channel 2
of the time capture file 'TEST.TIM', specify

          filtersdf (b, a, 'TEST.TIM', 0, 1);

Note   This MEX-file requires the MATLAB Signal Processing Toolbox, because it uses the
☝   'filter' function.

---

matdemo.tim

This file contains data that can be used by the tcdemo.m M-file. The data represents a 4 channel, 12,800 Hertz , baseband, time capture with broadband noise on all 4 channels.

tcdemo.m

This M-file is a demonstration. It shows you how to digitally filter HP3566A/67A time-capture data using the MATLAB Signal Processing Toolbox.

This demonstration assumes that matdemo.tim exists in \analyzer\tc, and that it is

- A 4 channel time-capture file, with
- a span of 12,800 Hertz.

Note	This demonstration overwrites the data in matdemo.tim with refiltered data. Therefore, make a backup copy of matdemo.tim if you want to preserve its contents.

tcdemo2.m

This M-file is a demonstration. It shows you how to compare the FRF measurement results of the digitally filtered, HP3566/67A time-capture data and the orginal filter shapes generated by the PC-MATLAB Signal Processing Toolbox.

This demonstration requires these files:

- \analyzer\data\matfrf.dat
- \analyzer\utils\sdftoml.exe

The file sdftoml.exe is provided with the SDF utilities; the file matfrf.dat is not. You must create matfrf.dat on your HP3566/67A. To create matfrf.dat, perform the following steps:

1. Load the matdemo.tim time-capture file into the HP 3566/67A.
2. Run a stable-average measurement.
3. Save the results to \analyzer\data\matfrf.dat (using Save Data).

Note   The file matdemo.tim is provided with the SDF utilities. It is a 4 channel, FRF time-capture file, created with a span of 12,800 Hertz.

Mathcad® Examples

Mathcad is a registered trademark of MathSoft, Inc., 201 Broadway, Cambridge MA 02139.

Overview

The Mathcad examples described in this chapter are in the mathcad directory. From within Mathcad, you can read and/or print the files in that directory for more detailed information about these examples.

The mathcad directory contains several example Mathcad 4.0 files that demonstrate how to create waveforms and filters for use with the HP89410A and HP89440A (including option AYA). Although designed for use with the HP894x0A, the general approach to using Mathcad data with SDF is similar for all instruments incorporating SDF capability.

Each of the following files is described briefly below:

- mathcad\signals\ofstsine.mcd
- mathcad\signals\gmsk.mcd
- mathcad\signals\qpsk.mcd
- mathcad\filters\gauss.mcd
- mathcad\filters\raiscos.mcd
- mathcad\filters\rootcos.mcd

Mathcad Files

mathcad\signals\ofstsine.mcd

This file contains an example of how to create a waveform that can be used with the HP894x0A arbitrary source. The complex waveform (that is, real and imaginary components) is a simple sinewave that can be offset from the center frequency of the measurement.

mathcad\signals\gmsk.mcd

This file contains an example of how to create a pulsed Gaussian MSK waveform for use with the HP894x0A arbitrary source. In addition to being pulsed, the signal can be modified to have imperfections such as IQ gain imbalance and origin offset.

mathcad\signals\qpsk.mcd

This file contains an example of how to create a pulsed QPSK waveform for use with the HP894x0A arbitrary source. The waveform is filtered using a root-raised cosine ISI filter.

mathcad\filters\gauss.mcd

This file contains an example of how to create a custom gaussian filter for use with the AYA (Digital demodulation) option for the HP894x0A Vector Signal Analyzer.

mathcad\filters\raiscos.mcd

This file contains an example of how to create a raised cosine filter for use with the AYA (Digital demodulation) option for the HP894x0A Vector Signal Analyzer.

mathcad\filters\rootcos.mcd

This file contains an example of how to create a root-raised cosine filter for use with the AYA (Digital demodulation) option for the HP894x0A Vector Signal Analyzer.

SDF to PC-MATLAB and MATRIXx examples

Introduction

The following examples demonstrate:

- How to move an HP 3563A (or HP 3562A) file to MATRIXx or PC-MATLAB file format.

- How to load a converted file into MATRIXx.

- How to load a converted file into PC-MATLAB.

HP 3563A to MATRIXx or PC-MATLAB file formats

1. Save measurement setup and data from the HP 3563A to a LIF disk.

2. Insert the disk into a drive attached to your PC. Use the command LIF SCAN to identify disk (simply type LIF SCAN from the directory which contains your LIF utilities). If your LIF disk is in a HP 9122 connected to your PC, the results will resemble:

```
 Drive
Specifier      Looking for Disk Drives
=========     =======================

  A:        Internal Flexible Disk
  B:        Internal Flexible Disk

           Found HP 88500A HP-IB Card at Select Code 7
              Found SS/80 Disk at Address 2,, HP 9122
  702:          Unit 0, Flexible or Removable Disk
  70201:         Unit 1, Flexible or Removable Disk
```

3. Do a directory of your disk, using the command LIF LS.

```
        LIF LS 702: /L
```

You'll see:

```
filename  type  start  size    implement   created
=========================================================== =========
SESTATE1  BDAT  155   4      20200080    89/08/03 16:16:08
DAFRF    BDAT  173  2      20200080    89/03/04 12:51:03
```

4. Copy the data file from LIF to your DOS home directory.

```
        LIF CP 702:DAFRF FRF
```

5. Convert data information to SDF format.

```
        63TOSDF FRF FRF.DAT
```

6. Copy data information to either MATRIXx or PCMATLAB format. Use the /X option to get X trace values. PCMATLAB files are binary, MATRIXx are ASCII. Either of these commands creates the file FREQRESP.MAT.

```
        SDFTOML FRF.DAT /X     (PC-MATLAB)
        SDFTOMX FRF.DAT /X     (MATRIXx)
```

Loading a File into PC-MATLAB

1. To **load** FREQRESP.MAT, start the PC-MATLAB program and type:

   load a:FREQRESP.MAT

2. To **get** a list of the variables that were loaded, type:

   who

   To **get** a list of the variables including size and whether it is complex, type:

   whos

   To **see** the X-axis values of data converted with the x option, type:

   o2i1x

   To **see** the Y-axis values, type:

   o2i1

3. To **plot** X and Y values on the screen (real data), type:

   plot(o2i1x,o2i1)

   To **plot** X and Y values on the screen (imaginary data), type:

   plot(o2i1x,imag(o2i1))

   To **display** dB on Y-axis:

   plot(o2i1x,20.*log10(abs(o2i1)))

   To **display** phase on Y-axis:

   plot(o2i1x,atan2(imag(o2i1),real(o2i1)))
   or
   plot(o2i1x,angle(o2i1))   angle() is an M-file in MATLAB TOOLBOX

   To **display** unwrapped phase on Y-axis:

   plot(o2i1x,unwrap(o2i1)) unwrap() is an M-file in MATLAB TOOLBOX

4. To **send** screen plots to an HP-GL plotter:

   After plotting on the screen, from inside PCMATLAB, type:

   meta FREQRESP (creates file FREQRESP.MET)

   From MS-DOS, type:

   gpp FREQRESP /dhpgl (creates HPGL plot file)

Loading a File into MATRIXx

1. To **load** FREQRESP.MAT, start MATRIXx and type:

    load ('FREQRESP.MAT')

2. To **see** X-axis values, type:

    o2i1x

    To **see** Y-axis values, type:

    o2i1

3. To **plot** x and y values on the screen (real data), type:

    plot(o2i1x, o2i1)

    To **plot** x and y values on the screen (imaginary data), type:

    plot(o2i1x,imag(o2i1))

    To **display** dB on Y-axis:

    plot(o2i1x,10*.434294482*log(o2i1.*conjg(o2i1)))

    To **display** phase on Y-axis:

    plot(o2i1x,atan(imag(o2i1)./real(o2i1)))

4. To **send** screen dumps to an HP-IB Printer:

    plot(o2i1x,o2i1,'PRINTER')

Standard Data Format (SDF)

This appendix describes the organization of files that use the Standard Data Format (SDF). SDF files contain both measurement data and the settings of important instrument parameters when the data was taken.

For more information on SDF data see chapter 2 and chapter 6.

Records—the Basic SDF Units

An SDF file consists of a number of smaller units called records: one "master," called the File Header, followed by a variable number of additional records. A record contains a group of related parameters and data. For example, a record called the Channel Header contains all channel-specific information for one of the channels used in a measurement.

There are eleven common records defined for SDF files:

File Header:	Provides an index to the file.
Measurement Header:	Contains settings of measurement parameters.
Data Header:	Tells you how reconstruct one block of measurement results (x- and y-axis values for every point of every trace).
Vector Header:	Tells you which channel (or pair of channels) provided data for a single trace.
Channel Header:	Contains channel-specific information for one channel used in the measurement.
Scan Structure:	Tells you how vectors are organized in the Y-axis Data record when the measurement includes multiple scans of data.
Scan Big:	Extended scan header which tells how vectors are organized in the Y-axis data record when the measurement may include more than 32767 scans of data.
Scan Variable:	Contains a number that identifies every scan (scan time, RPM, or scan number).
Comment Header:	Contains text information associated with the file.
X-axis Data:	Contains the x-axis data needed to reconstruct any trace.
Y-axis Data:	Contains the y-axis data needed to reconstruct any trace.

Another record—the Unique record—makes the SDF file format flexible. The eight common record types define parameters that are common to many instruments and systems. However, a particular instrument or system may need to save and recall the states of additional parameters. These states can reside in Unique records.

Record Order

Each measurement file begins with two ASCII characters followed by the File Header. Fields in the File Header tell you how many additional records follow and their order in the file. The order of records is always the same (although some files may not contain all record types).
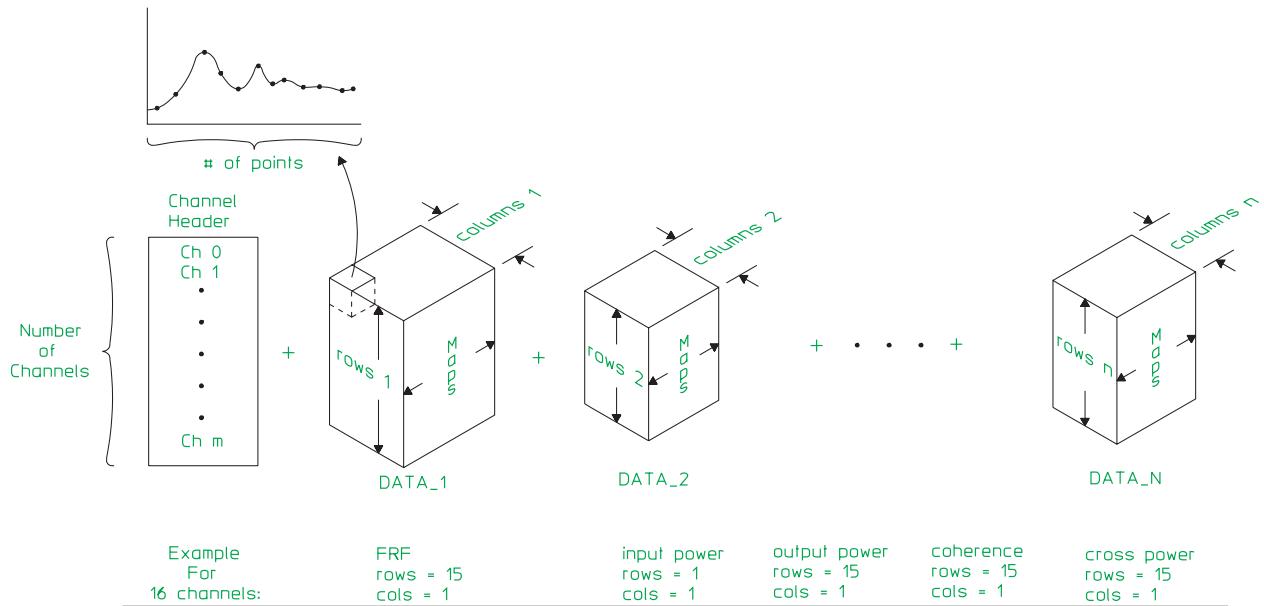

Figure B-5. Conceptual Model of an SDF File

(2) ASCII characters

B \0
SDF_FILE_HDR
SDF_MEAS_HDR
SDF_DATA_HDR
SDF_DATA_HDR
SDF_DATA_HDR
SDF_VECTOR_HDR
SDF_VECTOR_HDR
SDF_VECTOR_HDR
SDF_VECTOR_HDR
SDF_VECTOR_HDR
SDF_VECTOR_HDR
SDF_VECTOR_HDR
SDF_VECTOR_HDR
SDF_VECTOR_HDR
SDF_VECTOR_HDR
SDF_CHANNEL_HDR
SDF_CHANNEL_HDR
SDF_CHANNEL_HDR
SDF_CHANNEL_HDR
UNIQUE records
SDF_SCAN_STRUCT
SDF_SCAN_BIG
SDF_SCAN_VAR
SDF_SCAN_VAR
SDF_COMMENT_HDR
SDF_COMMENT_HDR
SDF_HDR (XDATA_TYPE)
SDF_HDR (YDATA_TYPE)

(1) File Header record — SDF_FILE_HDR

(1) Measurement Header record — SDF_MEAS_HDR

(0 or more) Data Header records — SDF_DATA_HDR

(0 or more) Vector Header records — SDF_VECTOR_HDR

(0 or more) Channel Header records — SDF_CHANNEL_HDR

(0 or more) Unique records — UNIQUE records

(0 or 1) Scan Structure records — SDF_SCAN_STRUCT

(0 or more) Scan Big records — SDF_SCAN_BIG

(0 or more) Scan Variable records — SDF_SCAN_VAR

(0 or more) Comment records — SDF_COMMENT_HDR

(0 or 1) X-axis Data records — SDF_HDR (XDATA_TYPE)

(0 or 1) Y-axis Data records — SDF_HDR (YDATA_TYPE)

Figure B-6. SDF File Structure

The two ASCII characters at the beginning of the file identify it as an SDF file. The first character is B, the second is \0 (null).

Special Fields in a Record

All records contain two special fields:

recordType: A number that uniquely identifies the type of data contained in the record.

recordSize: A number that specifies the length of the record (in bytes).

The following table shows you the recordType and recordSize values for the eight common records:

Table B-1. Common Records

Name	Description	recordType	SDF 1 recordSize (bytes)	SDF 2 recordSize (bytes)	SDF 3 recordSize (bytes)
SDF_FILE_HDR	File Header	10	64	64	80
SDF_MEAS_HDR	Measurement Header	11	102	140	156
SDF_DATA_HDR	Data Header	12	114	134	148
SDF_VECTOR_HDR	Vector Header	13	18	18	18
SDF_CHANNEL_HDR	Channel Header	14	146	192	212
SDF_SCAN_STRUCT	Scan Structure	15	Variable	Variable	Variable
SDF_XDATA_HDR	X-axis Data	16	Variable	Variable	Variable
SDF_YDATA_HDR	Y-axis Data	17	Variable	Variable	Variable
SDF_SCAN_BIG	Scan Big	18	N/A	N/A	20
SDF_SCAN_VAR	Scan Variable	19	N/A	N/A	Variable
SDF_COMMENT_HDR	Comment Header	20	N/A	N/A	Variable

Record and Structure Descriptions

The following tables describes the individual records and structures from which the SDF files are built. Each table includes:

- A field index.
- A binary index.
- A description of each field.
- A description of the data type used in each field.
- The range of values for data in the field.

The fields in SDF records and structures use the following data types:

Char[n]:   This data type consists of a series of ASCII-encoded bytes. [n] specifies the number of bytes in the field.

Short: This data type is used for integers (maximum range is –32768 to +32767). Values are transferred as two-byte, binary encoded integers.

Long: This data type is also used for integers (but the maximum range is –2,147,483,648 to +2,147,483,647). Values are stored as four-byte, binary-encoded integers.

Float: This data type is used for single-precision fixed-point and floating-point numbers. Values are stored in the 32-bit binary floating-point format.

Double: This data type is used for double-precision fixed-point and floating-point numbers. Values are stored using the 64-bit binary floating-point format.
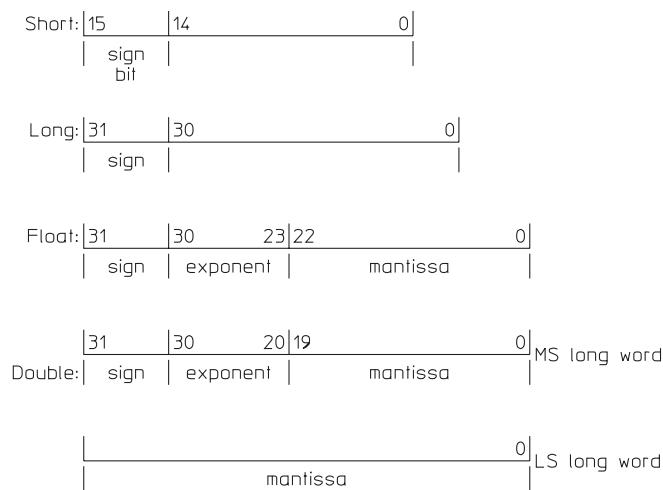
Figure B-7. IEEE Data Type Definition

Figure B-8. SDF Field Data Types

Table B-2. SDF_FILE_HDR (File Header record)

Field Index	Binary Index*	Field Name/Description	Data Type	Range/ Units
1	1:2	recordType	short	10
2	3:6	recordSize	long	80 bytes
3	7:8	revisionNum — measurement file version number.	short	0:32767
4	9:10	applic — file saved from this instrument or application. −1=HP VISTA −2=HP SINE −3=HP 35660A −4=HP 3562A, HP 3563A −5= HP 3588A −6= HP 3589A −99=unknown 1= HP 3566A, HP 3567A 2= HP 35665A 3= HP 3560A 4= HP 89410A, HP 89440A 7= HP 35635R 8= HP 35654A-S1A 9= HP 3569A 10= HP 35670A 11= HP 3587S	short	−99:32767
5	11:12	yearStamp — year at measurement start.	short	0:9999
6	13:14	monthDayStamp — month at measurement start. Encoded as (month $\times$ 100) + day For example, November 9 = 1109	short	0:1231
7	15:16	hourMinStamp — time at measurement start. Encoded as (hour $\times$ 100) + minute For example, 14:45 = 1445	short	0:2359
8	17:24	applicVer — software or firmware version number.	char[8]	i.d.
9	25:26	num_of_DATA_HDR_record — total Data Header records.	short	1:32767

* Bytes   i.d. = instrument-dependent

Table B-2. SDF_FILE_HDR (File Header record), continued

Field Index	Binary Index*	Field Name/Description	Data Type	Range/Units
10	27:28	num_of_VECTOR_record — total Vector Header records.	short	0:32767
11	29:30	num_of_CHANNEL_record — total Channel Header records.	short	0:32767
12	31:32	num_of_UNIQUE_record — total Unique records.	short	0:32767
13	33:34	num_of_SCAN_STRUCT_record — total Scan Structure records.	short	0:1
14	35:36	num_of_XDATA_record — total X-axis Data records.	short	0:1
15	37:40	offset_of_DATA_HDR_record — first Data Header record's byte offset from beginning of file.	long	$-1:(2^{31})-1$
16	41:44	offset_of_VECTOR_record — first Vector Header record's byte offset from beginning of file.	long	$-1:(2^{31})-1$
17	45:48	offset_of_CHANNEL_record — first Channel Header record's byte offset from beginning of file.	long	$-1:(2^{31})-1$
18	49:52	offset_of_UNIQUE_record — first Unique record's byte offset from beginning of file.	long	$-1:(2^{31})-1$
19	53:56	offset_of_SCAN_STRUCT_record — Scan Structure record's byte offset from beginning of file.	long	$-1:(2^{31})-1$
20	57:60	offset_of_XDATA_record — X-axis Data record's byte offset from beginning of file.	long	$-1:(2^{31})-1$
21	61:64	offset_of_YDATA_record — Y-axis Data record's byte offset from beginning of file.	long	$-1:(2^{31})-1$
22	65:66	num_of_SCAN_BIG_RECORD — total of SDF_SCAN_BIG and SDF_SCAN_VAR records.	short	0:32767
23	67:68	num_of_COMMENT_record — total of SDF_COMMENT_HDR records.	short	0:32767
24	69:72	offset_of_SCAN_BIG_record — the offset (from beginning of file) of the first Scan Big or Scan Variable record.	long	$-1:(2^{31})-1$
25	73:76	offset_of_next_SDF_FILE — allows more than one logical SDF FILE in a physical file. This supports multiple independent results taken at the same time. (For example, a time capture where the span and center frequencies of each channel are completely unrelated.) This offset points to the FORMAT_STRUCT record of the next logical SDF FILE in this physical file. All offsets in the next SDF FILE are relative to the start of the FORMAT field of the logical file (that is, ''B'' followed by ''\0'').	long	$-1:(2^{31})-1$

* Bytes   i.d. = instrument-dependent

* Bytes   i.d. = instrument-dependent

Table B-3a. SDF_MEAS_HDR (Measurement Header record)

Field Index	Binary Index*	Field Name/Description	Data Type	Range/ Units
1	1:2	recordType	short	11
2	3:6	recordSize	long	156 bytes

3	7:10	unique_record — byte offset from the beginning of the file to
		   a record containing an instrument-specific measurement
		   header. This field may be ignored when the file is recalled
		   if it is recalled into an instrument type other than that used
		   to create it.
			long	$-1:0:(2^{31})-1$

4	11:14	centerFreqOld** — center frequency.	float	unit is Hz, range is i.d.
5	15:18	spanFreqOld** — frequency span.	float	unit is Hz range is i.d.
6	19:22	blockSize — number of time-domain samples taken. This field is only valid for FFT measurements.	long	
7	23:24	zoomModeOn — zoom mode (0=not zoomed, 1=zoomed). This field is only valid for FFT measurements.	short	0:1
8	25:26	startFreqIndexOld*** — the first alias-protected point on a frequency-domain trace.	short	0:last_valid_index (SDF_DATA_HDR)
9	27:28	stopFreqIndexOld*** — the last alias-protected point on a frequency-domain trace.	short	0:last_valid_index (SDF_DATA_HDR)

10	29:30	averageType
		       0=none
		        1=rms
		       2=rms exponential
		        3=vector
		       4=vector exponential
		        5=continuous peak hold
		       6=peak
			short	0:6

11	31:34	averageNum — number of averages.	long	range is i.d.
12	35:38	pctOverlap — percentage of time-domain samples that are shared between successive time records. This field is only valid for FFT measurements.	float	number between 0 and 1
13	39:98	measTitle — measurement title or automath label.	char[60]	i.d.

14	99:102	videoBandWidth — tells you the bandwidth of the instrument's
		   video filter. This field is only valid for swept spectrum
		   measurements.
			float	unit is Hz

    * Bytes   ** Prior to version 2.0   *** Prior to version 3.0   i.d. = instrument-dependent

Table B-3b. SDF_MEAS_HDR (Measurement Header record)

Field Index	Binary Index*	Field Name/Description	Data Type	Range/Units
15	103:110	centerFreq — center frequency	double	unit is Hz range is i.d.
16	111:118	spanFreq — frequency span	double	unit is Hz range is i.d.
17	119:126	sweepFreq — current frequency for a swept measurement	double	unit is Hz range is i.d.

18	127:128	measType — measurement type
		      −99 = unknown measurement
		      0 = spectrum measurement
		      1 = network measurement
		      2 = swept measurement
		      3 = FFT measurement
		      4 = orders measurement
		      5 = octave measurement
		      6 = capture measurement
		      7 = correlation measurement
		      8 = histogram measurement
		      9 = swept network measurement
		      10 = FFT network measurement
			short	−99:10

19	129:130	realTime — whether the measurement was continuous in time
		      0 = not continuous
		      1 = continuous
			short	0:1

20	131:132	detection — detection type
		      −99 = unknown detection type
		      0 = sample detection
		      1 = positive peak detection
		      2 = negative peak detection
		      3 = rose-and-fell detection
			short	−99:3

21	133:140	sweepTime — actual time for a swept measurement	double	unit is sec range is i.d.
22	141:144	startFreqIndex — the first alias-protected point on a frequency-domain trace. This field is only valid for FFT measurements, long	long	0:last_valid_index (SDF_DATA_HDR)
23	145:148	stopFreqIndex — the last alias-protected point on a frequency-domain trace. This field is only valid for FFT measurements, long	long	0:last_valid_index (SDF_DATA_HDR)
24	149:156	expAverageNum — number of exponential averages.	double	range is i.d.

* Bytes   i.d. = instrument-dependent

Table B-4a. SDF_DATA_HDR (Data Header record)

Field Index	Binary Index*	Field Name/Description	Data Type	Range/ Units
1	1:2	recordType	short	12
2	3:6	recordSize	long	148 bytes

3	7:10	unique_record — byte offset from the beginning of the file to
		    a record containing an instrument-specific data header.
		    May be ignored if recalled into a different type instrument.
			long	$-1:(2^{31})-1$

4	11:26	dataTitle — instrument- or user-supplied name for data type.	char[16]	i.d.

5	27:28	domain
		          −99=unknown
		            0=frequency
		            1=time
		            2=amplitude
		            3=RPM
		            4=order
		            5=channel
		            6=octave
			short	−99, 0:6

6	29:30	dataType
		          −99=unknown
		            0=time
		            1=linear spectrum
		            2=auto-power spectrum
		            3=cross-power spectrum
		            4=frequency response
		            5=auto-correlation
		            6=cross-correlation
		            7=impulse response
		            8=ordinary coherence
		            9=partial coherence
		            10=multiple coherence
		            11=full octave
		            12=third octave
		            13=convolution
		            14=histogram
		            15=probability density function
		            16=cumulative density function,
		            17=power spectrum order tracking
		            18=composite power tracking
		            19=phase order tracking
		            20=rpm spectral
		            21=order ratio
		            22=orbit
		            23=HP 35650 series calibration
		            24=sine rms pwr data
		            25=sine variance data
		            26=sine range data
		            27=sine settle time data
		            28=sine integ time data
		            29=sine source data
		            30=sine overload data
		            31=sine linear data
		            32=synthesis
		            33=curve fit weighting function
		            34=frequency corrections (for capture)
		            35=all pass time data
		            36=norm reference data
			short	−99, 0:76

* Bytes   i.d. = instrument-dependent

Table B-4b. SDF_DATA_HDR (Data Header record)

Field Index	Binary Index*	Field Name/Description	Data Type	Range/ Units
6 (cont.)		dataType (continued) 37=tachometer data 38=limit line data 39=twelfth octave data 40=S11 data 41=S21 data 42=S12 data 43=S22 data 44=PSD data 45=decimated time data 46=overload data 47=compressed time data 48=external trigger data 49=pressure data 50=intensity data 51=PI index data 52=velocity data 53=PV index data 54=sound power data 55=field indicator data 56=partial power data 57=Ln 1 data 58=Ln 10 data 59=Ln 50 data 60=Ln 90 data 61=Ln 99 data 62=Ln user data 63=T20 data 64=T30 data 65=RT60 data 66=average count data 68= IQ measured time 69= IQ measured spectrum 70= IQ reference time 71= IQ reference spectrum 72= IQ error magnitude 73= IQ error phase 74= IQ error vector time 75= IQ error vector spectrum 76= symbol table data		
7	31:32	num_of_pointsOld***	short	
8	33:34	last_valid_indexOld***	short	
9	35:38	abscissa_firstXOld**	float	
10	39:42	abscissa_deltaXOld**	float	

* Bytes   ** Prior to version 2.0   *** Prior to version 3.0   i.d. = instrument-dependent

Table B-4c. SDF_DATA_HDR (Data Header record)

Field Index	Binary Index*	Field Name/Description	Data Type	Range/ Units
11	43:44	xResolution_type — tells you how to find x-axis values for this Data Header record's traces. 　　0=linear — calculate values from abscissa_firstX and abscissa_deltaX 　　1=logarithmic — calculate values from abscissa_firstX and abscissa_deltaX 　　2=arbitrary, one per file — find values in the X-axis Data record; same vector used for every trace in the measurement file 　　3=arbitrary, one per data type — find values in the X-axis Data record; same x-axis vector used for each trace associated with this record 　　4=arbitrary, one per trace — find values in the X-axis Data record; unique x-axis vector for each trace associated with this record	short	0:4
12	45:46	xdata_type — tells you the size and format of each x-axis value. 　　1=short (two-byte, binary-encoded integer) 　　2=long (four-byte, binary-encoded integer) 　　3=float (four-byte, binary floating-point number) 　　4=double (eight-byte, binary floating-point number) This field is only valid if xResolution_type is 2, 3, or 4.	short	1:4
13	47:48	xPerPoint — number of x-axis values per each trace point. This field is only valid if xResolution_type is 2, 3, or 4.	short	0:32767
14	49:50	ydata_type — tells you the size and format of each y-axis value. 　**NOTE**: If yIsComplex=1, both the real and imaginary components of each y value require the number of bytes specified here. 　　1=short (two-byte, binary-encoded integer) 　　2=long (four-byte, binary-encoded integer) 　　3=float (four-byte, binary floating-point number) 　　4=double (eight-byte, binary floating-point number)	short	1:4
15	51:52	yPerPoint — number of y-axis values per each trace point. 　**NOTE**: A value containing both real and imaginary components is still considered a single value.	short	0:32767
16	53:54	yIsComplex 　　0=each y value has only a real component 　　1=each y value has both a real and an imaginary component	short	0:1
17	55:56	yIsNormalized 　　0=not normalized 　　1=normalized (all y values fall between 0.0 and 1.0 and are unitless, for example coherence or power spectrum).	short	0:1
18	57:58	yIsPowerData 　　0=not power data (for example, linear spectrum) 　　1=power data (for example, auto-power spectrum)	short	0:1

　　*Bytes　　　　　　　　　　i.d. = instrument-dependent

Table B-4d. SDF_DATA_HDR (Data Header record)

Field Index	Binary Index*	Field Name/Description	Data Type	Range/ Units
19	59:60	yIsValid — (0=not valid, 1=valid)	short	0:1
20	61:64	first_VECTOR_recordNum — first Vector Header record belonging to this Data Header record	long	0:(num_of_ VECTOR_ record) − 1 (SDF_FILE_ HDR)
21	65:66	total_rows — used to determine the number of traces associated with this Data header record; just multiply total_rows by total_columns	short	1:32767
22	67:68	total_cols — used to determine the number of traces associated with this Data header record; just multiply total_rows by total_columns	short	1:32767
23	69:90	xUnit — engineering unit used for x-axis values.	struct	see SDF_UNIT
24	91:92	yUnitValid — yUnit field in this record is valid 0=use Channel Header record's engUnit field for y-axis units. 1=use this record's yUnit field for y-axis unit.	short	0:1
25	93:114	yUnit — engineering unit used for y-axis values.	struct	see SDF_UNIT
26	115:122	abscissa_firstX — x-axis value of first point. This field is only valid if xResolution_type is 0 or 1.	double	$\lceil 10^{34}$
27	123:130	abscissa_deltaX — spacing between x-axis points. $xn = x(n\text{-}1) + abscissa\_deltaX$ (xResolution_type 0) $xn = x(n\text{−}1) \rceil abscissa\_deltaX$ (xResolution_type 1) This field is only valid if xResolution_type is 0 or 1.	double	$\lceil 10^{34}$
28	131:132	scanData — indicates whether the data is scanned 0 = This data header is associated with non-scanned data 1= This data header is associated with the scan structure	short	0:1
29	133:134	windowApplied — indicates whether the windows indicated have already been applied to the data 0 = windows have not been applied 1 = windows have been applied	short	0:1
30	135:138	num_of_points — number of discrete points in each trace associated with this record.	long	$0:(2^{31})\text{−}1$
31	139:142	last_valid_index — last point containing valid data.	long	0:(num_of_points)−1
32	143:144	overSampleFactor — Usually 1 > 1 = the data has been low-pass filtered but not decimated.	short	1:32767
33	145:146	multiPassMode — ''Multi-pass'' refers to a mode where data for multiple frequency spans is interleaved. 0 = not multi-pass data 1..4 = multi-pass, corresponding the HP 3565 gate array modes 5 = future multi-pass mode	short 0:5	

| 34 | 147:148 | multiPassDecimations —
       > 0 = the number of decimations included in
           the multi-pass data. | short | 0:32767 |

    *Bytes   i.d. = instrument-dependent

Table B-5. SDF_VECTOR_HDR (Vector Header record)

Field Index	Binary Index*	Field Name/Description	Data Type	Range/ Units
1	1:2	recordType	short	13
2	3:6	recordSize	long	18 bytes
3	7:10	unique_record — byte offset from the beginning of the file to a record containing an instrument-specific vector header. May be ignored if recalled into a different type instrument.	long	$-1{:}(2^{31}){-}1$
4	11:14	the_CHANNEL_record[2] — tells you which channel or channels provided data for this trace. Each element of this array contains an index to a Channel Header record. An element refers to no channel if the index value is −1.	short (2x)	−1, 0:32767
5	15:18	pwrOfChan[2] — tells you what exponent was applied to corresponding channel data to create this trace. (For example, pwrOfChan[0] is the exponent applied to the_CHANNEL record[0]'s data.) record [0] for row channel Chan [0] " " record [1] for column channel Chan [1] " " **NOTE**: You must divide each pwrOfChan value by 48 to obtain the true value of the exponent.	short (2x)	0:32767

    *Bytes   i.d. = instrument-dependent

Table B-6a. SDF_CHANNEL_HDR (Channel Header record)

Field Index	Binary Index*	Field Name/Description	Data Type	Range/ Units
1	1:2	recordType	short	14
2	3:6	recordSize	long	212 bytes
3	7:10	unique_record — byte offset from the beginning of the file to a record containing an instrument-specific vector header. May be ignored if recalled into a different type instrument.	long	$-1{:}(2^{31}){-}1$
4	11:40	channelLabel — channel documentation	char[30]	i.d.
5	41:52	moduleId — location of channel in instrument.	char[12]	
6	53:64	serialNum — instrument (or module) serial number.	char[12]	
7	65:88	window	struct	see SDF_ WINDOW

8	89:90	weight
		        0=no weighting
		         1=A-weighting
		         2=B-weighting
		         3=C-weighting
			short	0:3

8	91:94	delayOld***	float	
9	93:96	range	float	unit is dBV, range is i.d. & includes overhead for scaling

10	99:100	direction
		         −9=−TZ
		         −8=−TY
		         −7=−TX
		         −3=−Z
		         −2=−Y
		         −1=−X
		         0=no direction specified
		         1=X
		         2=Y
		         3=Z
		         4=R (radial)
		         5=T (tangential — theta angle)
		         6=P (tangential — phi angle)
		         7=TX
		         8=TY
		         9=TZ
			short	0:9

11	101:102	pointNum — test point on device under test	short	0:32676
12	103:104	coupling (0=DC, 1=AC)	short	0:1
13	105:106	overloaded (0=no, 1=yes)	short	0:1
14	107:116	intLabel — label for the instrument's internal unit (such as V)	char[10]	i.d.
15	117:138	engUnit — engineering unit (EU) definition for this channel	struct	see SDF_UNIT
16	139:142	int2engrUnit — EU correction factor. Divide internal-unit data by this value to get EU data	float	$\int 10^{34}$ (except 0)

* Bytes   *** Prior to version 3.0   i.d. = instrument-dependent

Table B-6b. SDF_CHANNEL_HDR (Channel Header record)

Field Index	Binary Index*	Field Name/Description	Data Type	Range/ Units
17	143:146	inputImpedance — Input impedance	float	unit ohm, range i.d.
18	147:148	channelAttribute -99 = unknown attribute 0 = no attribute 1 = tach attribute 2 = reference attribute 3 = tach and reference attribute 4 = clockwise attribute	short	$-99{:}3$
19	149:150	aliasProtected 0 = data was not alias protected 1 = alias protected	short	0:1
20	151:152	digital 0 = analog input channel 1 = digital input channel	short	0:1
21	153:160	channelScale — see channelOffset below	double	unit is volts, range is i.d.
22	161:168	channelOffset — when the data type is ''short'' or ''long'' the following formula will convert the data to volts: Volts = channelOffset + (channelScale)Ydata)	double	unit is volts, range is i.d.
23	169:176	gateBegin — Gated sweep start time	double	unit is sec, range is i.d.
24	177:184	gateEnd — Gated sweep stop time	double	unit is sec, range is i.d.
25	185:192	userDelay — User specified input channel time delay or line length (not trigger delay)	double	unit is sec, range is i.d.
26	193:200	delay — amount of time between trigger event and start of data collection	double	unit is sec, range is i.d.
27	201:208	carrierFreq — carrier frequency for demodulated data	double	unit is Hz, range is i.d.
28	209:210	channelNumber — zero-based channel number	short	0:32767
29	211:212	channelModule — zero-based channel module	short	0:32767

*Bytes   i.d. = instrument-dependent

Table B-7. SDF_SCAN_STRUCT (Scan Structure record)

Field Index	Binary Index*	Field Name/Description	Data Type	Range/Units
1	1:2	recordType	short	15
2	3:6	recordSize	long	variable
3	7:8	num_of_scan — number of times the instrument collected a complete set of x- and y-axis vectors for all scan-based data types.	short	$1:(2^{15})-1$
4	9:10	last_scan_index — index of the last valid scan	short	0:(num_of_scan)−1

5	11:12	scan_type — tells you how the vectors from different scans are organized in the Y-axis Data record.
		   0=depth — all scans for the first data type's vectors followed by all scans for the second data type's vectors, and so on.
		   1=scan — all data type's vectors for the first scan followed by all data type's vectors for the second scan, and so on.
			short	0:1

6	13:14	scanVar_type — tells you the size and format of each scan variable value.
		   1=short (two-byte, binary-encoded integer)
		   2=long (four-byte, binary-encoded integer)
		   3=float (four-byte, binary floating-point number)
		   4=double (eight-byte, binary floating-point number)
			short	1:4

7	15:36	scanUnit — engineering unit used for scan variables	struct	see SDF_UNIT
		Scan variable values start here.		

*Bytes   i.d. = instrument-dependent

Table B-8. SDF_SCANS_BIG (Scan big record)

Field Index	Binary Index*	Field Name/Description	Data Type	Range/ Units
1	1:2	recordType	short	18
2	3:6	recordSize	long	20
3	7:10	unique_record — byte offset from the beginning of the file to a record containing an instrument-specific scan big header. May be ignored if recalled into a different type instrument.	long	$-1:(2^{31})-1$
4	11:14	num_of_scan — number of times the instrument collected a complete set of x- and y-axis vectors for all scan-based data types.	long	$-1:(2^{31})-1$
5	15:18	last_scan_index — index of the last valid scan.	long	0:(num_of_ scan)−1
6	19:20	scan_type — tells you how the vectors from different scans are organized in the Y-axis Data record. 0=depth — all scans for the first data type's vectors followed by all scans for the second data type's vectors, and so on. 1=scan — all data type's vectors for the first scan followed by all data type's vectors for the second scan, and so on.	short	0:1

Table B-9. SDF_SCAN_VAR (Scan variable record)

Field Index	Binary Index*	Field Name/Description	Data Type	Range/ Units
1	1:2	recordType	short	19
2	3:6	recordSize	long	variable
3	7:10	unique_record — byte offset from the beginning of the file to a record containing an instrument-specific scan variable header. May be ignored if recalled into a different type instrument.	long	$-1:(2^{31})-1$
4	11:14	headersize — size of the header portion of this record (excluding the scan variable values).	long	54

5	15:16	scanBase_type — type of scan variable
		    0 = unknown
		    1 = scan number
		    2 = time
		    3 = RPM
		    4 = temperature
		    5 = tachometer count
			short	0:5

6	17:18	scanOrder_type — progression of scan values
		    0 = unknown
		    1 = increasing in value
		    2 = decreasing in value
			short	0:2

7	19:20	DATA_recordNum— SDF_DATA_HDR record number associated with this record, (-1 if no specific association).	short	0:num_of_ DATA_HDR_ record -1
8	21:30	scan_ID — name of scan information.	char[10]	i.d.

9	31:32	scanVar_type — tells you the size and format of each scan variable value
		    1 = short (two-byte binary-encoded integer)
		    2 = long (four-byte binary-encoded integer)
		    3 = float (four-byte floating-point number)
		    4 = double (eight-byte floating-point number)
			short	1:4

10	33:54	scanUnit — engineering unit used for scan.	struct	see SDF_UNIT
		Scan variable values start here.		

Table B-10. SDF_COMMENT_HDR (Comment record)

Field Index	Binary Index*	Field Name/Description	Data Type	Range/ Units
1	1:2	recordType	short	20
2	3:6	recordSize	long	variable
3	7:10	unique_record — byte offset from the beginning of the file to a record containing an instrument-specific comment header. May be ignored if recalled into a different type instrument.	long	$-1{:}(2^{31}){-}1$
4	11:14	headersize — size of the header portion of this record (excluding the comment text).	long	24
5	15:18	comment_bytes — size of comment (in bytes). This size may be smaller than the comment text area. If the size of the text is −1, then the entire comment text area is valid (or until an end-of-text marker is found).	long	−1: recordSize− headerSize
6	19:20	comment_type — type of comment data 0 = contains text	short	0:0

7	21:22	scope_type — tells which type of header the comment applies to
		    0 = entire file
		    1 = SDF_DATA_HDR
		    2 = SDF_VECTOR_HDR
		    3 = SDF_CHANNEL_HDR
		    4 = SDF_SCAN_STRUCT
			short	0:4

8	23:24	scope_info — the index of the header associated with the scope_type (−1 = no specific header)	short	−1:32767
		Comment data starts here.		

Table B-11. SDF_XDATA_HDR (X-axis Data record)

Field Index	Binary Index*	Field Name/Description	Data Type	Range/ Units
1	1:2	recordType	short	16
2	3:6	recordSize	long	Variable
		X data values start here.		

*Bytes

Table B-12. SDF_YDATA_HDR (Y-axis Data record)

Field Index	Binary Index*	Field Name/Description	Data Type	Range/ Units
1	1:2	recordType	short	17
2	3:6	recordSize	long	Variable
		Y data values start here.		

*Bytes

Table B-13. SDF_UNIT (Unit structure)

Field Index	Binary Index*	Field Name/Description	Data Type	Range/ Units
1	1:10	label — engineering unit label (such as g).	char[10]	i.d.
2	11:14	factor — converts this unit to the equivalent SI unit	float	$\int 10^{34}$ (except 0)
3	15	mass — 2 times the unit exponent for the mass dimension	char	−128:127
4	16	length — 2 times the unit exponent for the length dimension	char	−128:127
5	17	time — 2 times the unit exponent for the time dimension	char	−128:127
6	18	current — 2 times the unit exponent for the current dimension	char	−128:127
7	19	temperature — 2 times the unit exponent for the temperature dimension	char	−128:127
8	20	luminal_intensity — 2 times the unit exponent for the luminal intensity dimension	char	−128:127
9	21	mole — 2 times the unit exponent for the mole dimension	char	−128:127
10	22	plane_angle — 2 times the unit exponent for the plane angle dimension	char	−128:127

*Bytes   i.d. = instrument-dependent

Table B-14. SDF_WINDOW (Window structure)

Field Index	Binary Index*	Field Name/Description	Data Type	Range/ Units

1	1:2	windowType
		    0=window not applied
		    1=Hanning
		    2=Flat Top
		    3=Uniform
		    4=Force
		    5=Response
		    6=user-defined
		    7=Hamming
		    8=P301
		    9=P310
		    10=Kaiser-Bessel
		    11=Harris
		    12=Blackman
		    13=Resolution filter
		    14=Correlation Lead Lag
		    15=Correlation Lag
		    16=Gated
		    17=P400
			short	0:12

2	3:4	windowCorrMode
		    0=correction not applied
		    1=narrow band correction applied
		    2=wide band correction applied
			short	0:2

3	5:8	windowBandWidth **NOTE**: When windowType = 13, this field contains the instrument's resolution bandwidth.	float	unit is bins unit is Hz, range is i.d.
4	9:12	windowTimeConst — determines decay of Force and Response windows	float	unit is sec, range is i.d.
5	13:16	windowTrunc — width of FORCE window	float	unit is sec, range is i.d.
6	17:20	wideBandCorr — correction factor for wide-band signals (like random noise)	float	$\pm 10^{34}$ (except 0)
7	21:24	narrowBandCorr — correction factor for narrow band signals (like sinesoidal wave)	float	$\pm 10^{34}$ (except 0)

    *Bytes   i.d. = instrument-dependent

Reconstructing a Trace

The simplest SDF file defines a single trace. A trace consists of a series of discrete points, each defined by an X-axis value and a Y-axis value. This section tells you how to locate or calculate a single trace's X-axis and Y-axis values.

X-axis Values

A trace's X-axis values are stored in one of two SDF records: the Data Header record, or the X-axis Data record. The location used depends on whether or not consecutive points are evenly spaced along the X-axis:

Evenly-spaced points: X-axis values are stored in the Data Header record.
Unevenly-spaced points: X-axis values are stored in the X-axis Data record.

Field 11 of the Data Header record (xResolution_type) is an indicator of X-axis spacing. Consecutive points are evenly spaced if xResolution_type has value of 0 or 1; they are unevenly spaced if it has a value of 2, 3, or 4.

When consecutive trace points are evenly spaced along the X-axis (xResolution_type = 0, or 1), the SDF file does *not* include an X-axis Data record. Instead, the X-axis values of these points are completely defined by three fields in the Data Header record:

abscissa_firstX: tells you the X-axis value of the first point.

abscissa_deltaX: tells you the X-axis spacing between points.

num_of_points: tells you how many discrete points the trace is composed of.

You can calculate X-axis values this way when xResolution_type is 0 (linear resolution):

$x0 = $ abscissa_firstX
$x1 = x0 + $ abscissa_deltaX
$x2 = x0 + $ abscissa_deltaX $ * 2$
$x3 = x0 + $ abscissa_deltaX $ * 3$
   .
    .
    .
$xn = x0 + $ abscissa_deltaX $ * n$ (where $n = $ num_of_points $ - 1$)

You can calculate X-axis values this way when xResolution_type is 1 (logarithmic resolution):

$x0 = $ abscissa_firstX
$x1 = x0 * $ abscissa_deltaX
$x2 = x0 * $ abscissa_deltaX^2
$x3 = x0 * $ abscissa_deltaX^3
.
.
.
$xn = x0 * $ abscissa_deltaX ^{n} (where $n = $ num_of_points -1)

When consecutive trace points are *not* evenly spaced along the X-axis (xResolution_type = 2, 3, or 4), the SDF file includes an X-axis Data record. And when the file defines a single trace, that record includes just one vector of X-axis values.

Three fields in the Data Header record tell you how to read X-axis values from the X-axis Data record:

num_of_points: tells you how many discrete points the trace is composed of.

xdata_type: tells you the size and format of each X-axis value. (This information is encoded. For example, if xdata_type = 3, the size is four bytes and the format is binary floating-point.)

xPerPoint: tells you how many X-axis values are used to define a single point.

You can determine the size (in bytes) of the X-axis Data record's data block with the following formula:

  num_of_points * xPerPoint * Size

The value of "Size" is encoded in xdata_type, as shown below:

xdata_type	Size (bytes)
1	2
2,3	4
4	8

Y-axis Values

A trace's Y-axis values are always stored in an SDF file's Y-axis Data record. And when the file defines a single trace, that record includes just one vector of Y-axis values. Reading this vector of values is similar to reading a vector of X-axis values from the X-axis Data record.

Four fields in the Data Header record tell you how to read Y-axis values from the Y-axis Data record:

num_of_points: tells you how many discrete points the trace is composed of.

ydata_type: tells you the size and format of each Y-axis value. (This information is encoded. For example, if ydata_type = 3, the size is our bytes and the format is binary floating-point.)

yPerPoint: tells you how many Y-axis values are used to define a single point.

yIsComplex: tells you whether each Y-axis value is purely real, or whether it consists of a real and an imaginary component.

You can determine the size (in bytes) of the Y-axis Data record's data block with the following formula:

$$\text{num\_of\_points} \rangle \text{yPerPoint} \rangle \text{Size} \rangle 2^{(\text{yIsComplex})}$$

The value of "Size" is encoded in ydata_type, as shown below:

ydata_type	Size (bytes)
1	2
2,3	4
4	8

Correcting Y-axis Data

Y-axis data is stored in the instrument's internal unit (usually volts) with all calibration corrections applied. Two other kinds of correction, however, are left for you to apply:

- Engineering unit (EU) correction.
- Window correction. (Necessary only for FREQ or ORDER domain data)

Note

Do not apply a window correction factor if it has already been applied by the instrument. A field in the Channel Header record — windowCorrMode — tells you whether or not the instrument has applied a window correction factor.

Also, do not apply a window correction factor if you are using the SdfYdata function; it will automatically apply these corrections to the data. See the chapter ''Using SDF Data with C Programs'' for more information on SdfYdata.

EU correction allows you to convert y-axis data from the instrument's internal unit to some user-defined unit (such as g — the acceleration of gravity). An EU correction factor is included in each Channel Header record; the factor's field name is int2engrUnit.

Window correction compensates for the shape of the resolution bandwidth filter in swept-tuned measurements. It compensates for the frequency-domain shape of the windowing function in FFT-based measurements. (FFT, which stands for Fast Fourier Transform, is an implementation of the Discrete Fourier Transform. Both are math algorithms that transform data from the time domain to the frequency domain.)

Two window correction factors are included in each Channel Header record's SDF_WINDOW structure:

wideBandCorr: corrects for lost power when you are analyzing wide-band signals
                             (like random noise).

narrowBandCorr: corrects for lost amplitude when you are analyzing narrow-band signals
                             (like a sinesoidal wave).

Check the windowCorrMode field (also in SDF_WINDOW) to find out if your instrument has already applied one of these factors to the Y-axis data. The field contains encoded values:

       0:                   no window correction.

       1:                   narrowBandCorr.

       2:                   wideBandCorr.

Role of the Vector Header

The Vector Header record contains the information you need to link a trace to the appropriate correction factors. First, it tells you which channels were used to gather raw data for the trace. It is the correction factor for these channels that you will apply to the trace. Second, it tells you how the raw channel data was processed to create the trace data. The correction factors must be processed in the same way before they are applied to the trace data. An example will help to illustrate this point.

A frequency response trace is created by processing the data from two channels in the following way:

$$\text{frequency response} = \frac{\text{response channel spectrum}}{\text{excitation channel spectrum}}$$

This formula can also be written as:

$$\text{freq resp} = (\text{resp ch spec})^1 \times (\text{exci ch spec})^{-1}$$

A similar formula must be used to derive the correction factor for the frequency response trace:

$$\text{freq resp correction} = (\text{resp ch correction})^1 \times (\text{exci ch correction})^{-1}$$

What values would a Vector Header record contain for this frequency response trace? Here is an example record:

recordType:	13
recordSize:	18
unique_record:	-1
the_CHANNEL_record[0]:	5
the_CHANNEL_record[1]:	2
pwrOfChan[0]:	48
pwrOfChan[1]:	-48

Two fields in the record — the_CHANNEL_record[0] and the_CHANNEL_record[1] — tell you which channels were used to measure response and excitation. In the example used here, these values indicate that the channels are described in Channel Header record 5 and Channel Header record 2. Two other fields — pwrOfChan[0] and pwrOfChan[1] — tell you what exponent was applied to each channel's data. (pwrOfChan[0] provides the exponent for the_CHANNEL_record[0] and pwrOfChan[1] provides the exponent for the_CHANNEL_record[1].)

Note   You must divide each pwrOfChan value by 48 to obtain the true value of the exponent.

Numbers in the_CHANNEL_record fields are indexes to Channel Header records in the file (0 refers to the first one, 1 to the second, and so on). However, there is *no* correlation between a record's index number and the number of the channel described in that record. Channel Header record 0, for example, could describe Channel 2 of your instrument. To determine which channel is described in the record, you must read the record's channelLabel field.

Creating the Correction Factor

As you have seen in this section, the Vector Header record and its associated Channel Header records contain all the information you need to create a correction factor for a trace's Y-axis data. Just complete the following steps:

1. Determine which channels were used to gather data for the trace. (Look at the Vector Header record's the_CHANNEL_record fields.)
2. Determine which window correction factor (if any) was used for each channel. (Look in the Channel Header record's SDF_WINDOW structure; the field is windowCorrMode.)
3. Create a combined correction factor for each channel using the following formula:

$$\left\{ \frac{\text{window correction}}{\text{EU correction}} \right\}^{\text{pwrOfChan}/48}$$

   window correction is equal to 1 if step 2 indicates that no window correction was used, otherwise it is the value of either narrowBandCorr or wideBandCorr.

   EU correction is the value of int2engrUnit.
4. Create a trace correction factor by multiplying the two channel correction factors.

Now you can correct the Y-axis data. Just multiply each Y-axis value by the trace correction factor.

Note   Some traces are created with data from just one channel. (To indicate that only one channel was used, one "the_CHANNEL_record" field contains a value of -1.) For traces such as these, the combined correction factor for the single channel is also your trace correction factor.

Locating a Vector of Y-axis Data

When an SDF file defines just one trace, the Y-axis Data record contains a single vector of Y-axis data. However, when the file defines more than one trace, the record contains many vectors — one for each trace. This section tells you how to locate one Y-axis vector in a block of many.

Single-Scan Files

When a file does *not* contain a Scan Structure record, the structure of the Y-axis data block is quite simple: the order of vectors in the block is exactly the same as the order of Vector Header records in the file. The vector described by Vector Header record 0 is first in the block, the one described by Vector Header record 1 is second, and so on.

Multiple-Scan Files

When a file does contain a Scan Structure record, the structure of the Y-axis data block is somewhat more complex:

- The block contains more than one set vectors.
- The vectors can be organized in two different ways.

A single scan puts a complete set of vectors in the Y-axis data block — one for each Vector Header record. Each additional scan adds another set of vectors to the block.

The two types of vector organization are referred to as Scan and Depth. The scan_type field in the Scan Structure record tells you which type of organization was used for your file. The following example shows how Scan and Depth differ.

The example file contains two Data Header records and seven Vector Header records (num_of_DATA_HDR_record and num_of_Vector_record in the File Header record have values of 2 and 7, respectively). Its Y-axis data block includes three scans of data (num_of_scan in the Scan Structure record has a value of 3). The first Data Header record is linked to the first three Vector Header records (total_rows �txtotal_cols = 3; first_VECTOR_recordNum = 0) and the second is linked to the last four (total_rows �txtotal_cols = 4; first_VECTOR_recordNum = 3).

Here are the two ways the vectors could be organized in the Y-axis data block:

scan_type is Scan (1):		scan_type is Depth (0):	
S0.D0.	V0	S0.D0.	V0
	V1		V1
	V2		V2
D1.	V3	S1.D0.	V0
	V4		V1
	V5		V2
	V6	S0.D1.	V3
S1.D0.	V0		V4
	V1		V5
	V2		V6
D1.	V3	S1.D1.	V3
	V4		V4
	V5		V5
	V6		V6

Time Capture Data

A time capture data contains long streams of raw time data captured by an instrument. The files contains some or all of the following results.

Time Data

Time data is identified in a time capture file with the following SDF_DATA_HDR attributes:

dataType	0 (time data)
ydata_type	SHORT16 or LONG32
scanData	1

The time data should always be the first result in a time capture file. To convert the data from its integer form to volts, you need to determine the channel header that belongs to the data and get the scaling factor from the channel header.

The time capture data is stored with multiple rows of data, one row for each channel of captured data. For a given row, the desired SDF_VECTOR_HDR number is (*first_VECTOR_recordNum* + row). After locating the SDF_VECTOR_HDR for this row, find the channel number from the_CHANNEL_record[0]. After locating the SDF_CHANNEL_HDR, you can scale the time capture data to volts by using the scale values in the SDF_CHANNEL_HDR. For each time capture value in the data result (defined as time_data[i]):

$$time\_volts[i] = channelOffset + (channelScale \times time\_data[i])$$

To determine the total number of time capture points, you need to know now many scans are in the file. The number of usable scans can be determined from the *last_scan_index* field in the SDF_SCAN_STRUCT. The number of usable points per scan can be determined from the last_valid_index in the SDF_DATA_HDR. The total number of time capture data points is:

$$time\_capture\_points = (last\_valid\_index+1) \times (last\_scan\_index+1)$$

Overload Data

Overload data is identified in a time capture file with the following SDF_DATA_HDR attributes:

dataType	46 (overload data)
ydata_type	SHORT16
num_of_points	1
scanData	1

This data result identifies whether an overload occurred in each scan of a time capture. There is one of these results for each channel of a time capture.

Frequency Correction Data

Frequency Correction Data is identified in a time capture file with the following SDF_DATA_HDR attributes:

dataType	34 (frequency corrections)
scanData	0

This data result contains the frequency domain corrections for the time capture data. It is the same size as the instrument's Spectrum data result. There is one of these results for each channel of a time capture.

Compressed Time Data

The compressed time data result in a time capture file contains a compressed representation of the entire time capture file of each channel of time data. The time data is sampled and minimum and maximum data values for each sampled region are stored for each data point. The compressed time data is identified in a time capture file with the following SDF_DATA_HDR attributes:

dataType	45 (decimated time data) or 47 (compressed time data)
ydata_type	SHORT16 or LONG32
yPerPoint	5
yIsComplex	0
scanData	0

The 5 values per point represent the following:

minimum real value

minimum imaginary value

maximum real value

maximum imaginary value

overload flag

Since the minimum/maximum values are in integer form, they can be scaled to volts using the same mechanism as for scaling the time data (see above). If the dataType is 45 (decimated time data), then the minimum values equal the maximum values since the data is decimated, not compressed. There is one of these results for each channel of a time capture.

Tachometer Data

Tachometer data is identified in a time capture file with the following SDF_DATA_HDR attributes:

dataType	37 (tachometer data)
ydata_type	LONG32
yIsComplex	0
scanData	0

Each point in the tachometer data is a 32 tachometer counter value which represents time. To interpret the tachometer data, you need to know how many tachometer points are in the tachometer data, the tachometer frequency, and the number of tachometer pulses per revolution.

$$number\_of\_tach\_points = (total\_rows \setminus total\_cols - 1) \setminus num\_of\_points + last\_valid\_index$$

$$tach\_pulses\_per\_rev = abscissa\_deltaX$$

$$tach\_frequency = abscissa\_firstX$$

For each tachometer counter value in the tachometer data result (defined as tach_cnt[i]):

$$tach\_time\_sec[i] = tach\_cnt[i] / tach\_frequency$$

From each consecutive pair of tachometer points, you can compute and instantaneous RPM:

$$tach\_pulse\_delta\_sec = tach\_time\_sec[i] - tach\_time\_sec[i - 1]$$

$$tach\_pulses\_per\_min = 60 \setminus tach\_pulses\_per\_sec$$

$$tach\_rpm = tach\_pulses\_per\_min / tach\_pulses\_per\_rev$$

There may be a delay between the tachometer zero count value (when the tachometer started counting) and the first sample in the time capture data. From the SDF_VECTOR_HDR associated with the tachometer data result, you can locate the corresponding SDF_CHANNEL_HDR. The delay, in seconds, is in the *userDelay* field.

There is only one tachometer result in a time capture.

External Trigger Data

External trigger data is identified in a time capture file with the following SDF_DATA_HDR attributes:

dataType	48 (external trigger data)
ydata_type	LONG32
yIsComplex	0
scanData	0

If the trigger mode is set to be the external trigger at the time the capture was acquired and the tachometer is turned on, then this data result may be present.

Each point in the external trigger data is a 32 external trigger counter value which represents time. There is one time value associated with each scan of the time capture. The external trigger counter value is converted to a time value the same way as the tachometer data (see above).

Waterfall/Map/MultiSpectrum/Spectrogram Data

Many instrument create multiple scans of data and display them as a waterfall, a map, a multispectrum, or a spectrogram. In an SDF data file, this appears as multiple scans of a data result.

When this type of data is in the file, the *scanData* field in the SDF_DATA_HDR for a result is set to TRUE (1) and the SDF_FILE _HDR indicates that either an SDF_SCAN_STRUCT record or an SDF_SCAN_BIG record is in the file (or both). The *last_scan_index* field in either of these records indicates how many scans of data are in the file.

A scan variable may be:

- A time offset from when the measurement was started
- A scan offset from when the measurement was started
- The RPM when the scan was measured
- The average count when the scan was measured
- The overload when the scan was measured
- Information flags when the scan was measured

The first scan variable is in the SDF_SCAN_STRUCT record. The rest of the scan variables are in SDF_SCAN_VAR records. Most instruments support only one scan variable, which is in the SDF_SCAN _STRUCT record.

The HP 89410A and HP 89440A support 4 scan variables as follows:

Scan variable	Description
0 (SDF_SCAN_STRUCT)	Scan offset from when the measurement was started
1 (SDF_SCAN_VAR)	Same as above
2 (SDF_SCAN_VAR)	The average count when the scan was measured
3 (SDF_SCAN_VAR)	Information flags when the scan was measured (see below)

Information flags for scan variable 3:

Bit	Meaning when bit is set
0	row channel overload
1	column channel overload
8	questionable data
9	uncalibrated data
15	real-time data
16	digital demodulation sync search is on
17	digital demodulation pulse search is on
24	digital demodulation sync found (only valid if sync search is on)
25	digital demodulation pulse found (only valid if pulse search is on)

Example File

The following multi-page illustration shows the contents of a simple measurement file (of the SDF 1.) It defines just one trace. The trace's Y-axis values are contained in a vector of data at the end of the file (in the record SDF_YDATA_HDR). The trace's X-axis values are calculated from information in the record SDF_DATA_HDR.

File power.dat
B

SDF_FILE_HDR

recordType:	10
recordSize:	64
revisionNum:	0
applic:	1
yearStamp:	0
monthDayStamp:	0
hourMinStamp:	0
applicVer:	
num_of_DATA_HDR_record:	1
num_of_VECTOR_record:	1
num_of_CHANNEL_record:	1
num_of_UNIQUE_record:	0
num_of_SCAN_STRUCT_record:	0
num_of_XDATA_record:	0
offset_of_DATA_HDR_record:	168
offset_of_VECTOR_record:	1532
offset_of_CHANNEL_record:	1550
offset_of_UNIQUE_record:	−1
offset_of_SCAN_STRUCT_record:	−1
offset_of_XDATA_record:	−1
offset_of_YDATA_record:	1692

SDF_MEAS_HDR

recordType:	11
recordSize:	102
unique_record:	−1
centerFreq:	2.560000E+04
spanFreq:	6.553600E+04
blockSize:	1024
zoomModeOn:	0
startFreqIndex:	0

stopFreqIndex:	400
averageType:	1
averageNum:	10
pctOverlap:	0.000000E+00
measTitle:	
videoBandWidth:	0.000000E+00

SDF_DATA_HDR 0

recordType:	12
recordSize:	114
unique_record:	−1
dataTitle:	Power Spectrum
domain:	0
dataType:	2
num_of_points:	513
last_valid_index:	512
abscissa_firstX:	0.000000E+00
abscissa_deltaX:	1.280000E+02
xResolution_type:	0
xdata_type:	3
xPerPoint:	0
ydata_type:	3
yPerPoint:	1
yIsComplex:	0
yIsNormalized:	0
yIsPowerData:	1
yIsValid:	1
first_VECTOR_recordNum:	0
total_rows:	1
total_cols:	1
unitLabel:	Hz
unitFactor:	6.283190E+00

0

0

−2

0

0

0

0

2

yUnitValid:	0
unitLabel:	
unitFactor:	0.000000E+00

0
0
0
0
0
0
0
0

SDF_VECTOR_HDR 0

recordType:	13
recordSize:	18
unique_record:	−1
the_CHANNEL_record[0]:	0
the_CHANNEL_record[1]:	−1
pwrOfChan[0]:	96
pwrOfChan[1]:	0

SDF_CHANNEL_HDR 0

recordType:	14
recordSize:	146
unique_record:	−1
channelLabel:	Chan 1
moduleId:	
serialNum:	P46.464646
windowType:	1
windowCorrMode:	0
windowBandWidth:	1.500000E+00
windowTimeConst:	0.000000E+00
windowTrunc:	0.000000E+00
wideBandCorr:	1.632993E+00
narrowBandCorr:	2.000000E+00
weight:	0
delay:	0.000000E+00
range:	2.000000E+00
direction:	3
pointNum:	1
coupling:	1
overloaded:	0
intLabel:	V
unitLabel:	g
unitFactor:	9.806650E+00

0

2

–4

0

0

0

0

0

| int2engrUnit: | 1.000000E-02 |
| inputImpedance: | 1.000000E+06 |

SDF_YDATA_HDR

recordType:	17
recordSize:	2058
7.688535E-03:	513 real points
1.922215E-03:	|
9.141845E-10:	V

.

.

.

2.377593E-12

C

Batch Files

Introduction

This chapter documents the batch files included with the Standard Data Format Utilities. These files ''automate'' some procedures by performing the two or three steps required. For example, ''63_SDF.BAT'' performs both the LIF-to-DOS conversion and the HP 3563A-to-SDF conversion for all HP 3563A/62A files on the specified drive.

You can use the batch files as they are, or copy and modify them to fit your particular needs.

The batch files included are:

63_ML.BAT	60_ML.BAT	69_ML.BAT	IB63_ML.BAT
63_MX.BAT	60_MX.BAT	69_MX.BAT	IB63_MX.BAT
63_PLOT.BAT	60_PLOT.BAT	69_PLOT.BAT	IB63_PLT.BAT
63_SDF.BAT	60_SDF.BAT	69_SDF.BAT	IB63_SDF.BAT

63_ML.BAT

Convert all HP 3563A/62A data files on the specified LIF disk drive to PC-MATLAB Mat-file format. Destination is *.MAT where ''*'' is the same name as the LIF file on the source disk drive. Default drive is ''b:''.

Syntax   63_ML [drive:]

```
@echo off

set drv63=%1
set dir63=tmp63
set dirsdf=tmpsdf
set file63=*.DAT
set tmp63=%dir63%\%file63%
set tmpsdf=%dirsdf%\%file63%
set tmpmat=%dirsdf%\*.MAT

if '%drv63%'=='' set drv63=b:
mkdir %dir63%
if exist %tmp63% del %tmp63%
mkdir %dirsdf%
if exist %tmpsdf63% del %tmpsdf%

echo Copy HP 3563A/62A LIF data files from %drv63%
lif cp %drv63%DA* %tmp63% /o

echo Convert HP 3563A/62A data files to SDF
chdir %dir63%
for %%f in (%file63%) do 63tosdf %%f ..\%dirsdf%\%%f /o
chdir ..

echo Convert SDF files to PC-MATLAB Mat-files
chdir %dirsdf%
 for %%f in (%file63%) do sdftoml %%f /b /o >nul
echo Copy PC-MATLAB Mat-files to current directory
copy *.MAT ..
chdir ..

del %tmpsdf%
del %tmpmat%
rmdir %dirsdf%
del %tmp63%
rmdir %dir63%

set drv63=
set dir63=
set dirsdf=
set tmp63=
set tmpsdf=
set tmpmat=
```

63_MX.BAT

Convert all HP 3563A/62A data files on the specified LIF disk drive to MATRIXx Mat-file format. Destination is *.MAT where ''*'' is the same name as the LIF file on the source disk drive. Default drive is ''b:''.

Syntax   63_MX [drive:]

```
@echo off

setdrv63=%1
set dir63=tmp63
set dirsdf=tmpsdf
set file63=*.DAT
set tmp63=%dir63%\%file63%
set tmpsdf=%dirsdf%\%file63%
set tmpmat=%dirsdf%\*.MAT

if '%drv63%'=='' set drv63=b:
mkdir %dir63%
if exist %tmp63% del %tmp63%
mkdir %dirsdf%
if exist %tmpsdf63% del %tmpsdf%

echo Copy HP 3563A/62A LIF data files from %drv63%
lif cp %drv63%DA* %tmp63% /o

echo Convert HP 3563A/62A data files to SDF
chdir %dir63%
for %%f in (%file63%) do 63tosdf %%f ..\%dirsdf%\%%f /o
chdir ..

echo Convert SDF files to MATRIXx Mat-files
chdir %dirsdf%
for %%f in (%file63%) do sdftomx %%f /b /o >nul
echo Copy MATRIXx Mat-files to current directory
copy *.MAT ..
chdir ..

del %tmpsdf%
del %tmpmat%
rmdir %dirsdf%
del %tmp63%
rmdir %dir63%

set drv63=
set dir63=
set dirsdf=
set tmp63=
set tmpsdf=
set tmpmat=
```

63_PLOT.BAT

Plot all HP 3563A/62A data files on the specified LIF disk drive. Default drive is ''b:''.

Syntax
63_PLOT [drive:]

```
@echo off

set drv63=%1
set dir63=tmp63
set file63=*.63
set tmp63=%dir63%\%file63%

if '%drv63%'=='' set drv63=b:
mkdir %dir63%
if exist %tmp63% del %tmp63%

echo Copy HP 3563A/62A LIF data files from %drv63%
lif cp %drv63%DA* %tmp63% /o

REM plot all *.63 files
chdir %dir63%
for %%f in (%file63%) do viewdata %%f /t:%%f /p
chdir ..

del %tmp63%
rmdir %dir63%

set drv63=
set dir63=
set file63=
set tmp63=
```

63_SDF.BAT

Convert all HP 3563A/62A data files on the specified LIF disk drive to SDF. Default drive is ''b:''.

Syntax 63_SDF.BAT [drive:]

```
@echo off

set drv63=%1
set dir63=tmp63
set file63=*.DAT
 set tmp63=%dir63%\%file63%

if '%drv63%'=='' set drv63=b:
mkdir %dir63%
if exist %tmp63% del %tmp63%

echo Copy HP 3563A/62A LIF data files from %drv63%
lif cp %drv63%DA* %tmp63% /o

chdir %dir63%
echo Convert HP 3563A/62A data files to SDF
for %%f in (%file63%) do 63tosdf %%f ..\%%f /o
chdir ..

del %tmp63%
rmdir %dir63%

set drv63=
set dir63=
set file63=
set tmp63=
```

60_ML.BAT

Convert all HP 3560A data files that are transferred over the RS-232 port to PC-MATLAB Mat-file format. The default port is ''1'' (COM1). The default baud rate is ''9600''.

Syntax   60_ML.BAT [port] [baud]

```
@echo off

set port60=%1
set baud60=%2
set temp60=temp60
set dire60=tempsdf

if '%port60%'=='' set port60=1
if '%baud60%'=='' set baud60=9600

mkdir %dire60%
chdir %dire60%
echo y | del %dire60%

download /o %temp60%.x32 /p%port60% /b%baud%

x32tobin /o %temp60%.x32 %temp60%.bin
del %temp60%.x32

60tosdf  /o %temp60%.bin
del %temp60%.bin

for %%f in (*.dat) do sdftoml %%f /b /o >nul
copy *.mat ..
chdir ..

echo y | del %dire60%
 rmdir %dire60%

set port60=
set baud60=
set temp60=
set dire60=
```

60_MX.BAT

Convert all HP 3560A data files that are transferred over the RS-232 port to MATRIXx Mat-file format. The default port is ''1'' (COM1). The default baud rate is ''9600''.

Syntax   60_MX.BAT [port] [baud]

```
@echo off

set port60=%1
set baud60=%2
set temp60=temp60
set dire60=tempsdf

if '%port60%'=='' set port60=1
if '%baud60%'=='' set baud60=9600

mkdir %dire60%
chdir %dire60%
echo y | del %dire60%

download /o %temp60%.x32 /p%port60% /b%baud%

x32tobin /o %temp60%.x32 %temp60%.bin
del %temp60%.x32

60tosdf  /o %temp60%.bin
del %temp60%.bin

for %%f in (*.dat) do sdftomx %%f /b /o >nul
copy *.mat ..
chdir ..

echo y | del %dire60%
rmdir %dire60%

set port60=
set baud60=
set temp60=
set dire60=
```

60_PLOT.BAT

Plot all HP 3560A data files that are transferred over the RS-232 port. The default port is ''1'' (COM1).
The default baud rate is ''9600''.

Syntax   60_PLOT.BAT [port] [baud]

```
@echo off

set port60=%1
 set baud60=%2
set temp60=temp60
set dire60=tempsdf

if '%port60%'=='' set port60=1
if '%baud60%'=='' set baud60=9600

mkdir %dire60%
chdir %dire60%
echo y | del %dire60%

download /o %temp60%.x32 /p%port60% /b%baud%

x32tobin /o %temp60%.x32 %temp60%.bin
del %temp60%.x32

60tosdf  /o %temp60%.bin
del %temp60%.bin

for %%f in (*.dat) do viewdata %%f /t:%%f /p >nul
chdir ..

echo y | del %dire60%
rmdir %dire60%

set port60=
set baud60=
set temp60=
set dire60=
```

60_SDF.BAT

Convert all HP 3560A data files that are transferred over the RS-232 port to SDF. The default port is ''1'' (COM1). The default baud rate is ''9600''.

Syntax   60_SDF.BAT [port] [baud]

```
@echo off

set port60=%1
set baud60=%2
set temp60=temp60

if '%port60%'=='' set port60=1
if '%baud60%'=='' set baud60=9600

download /o %temp60%.x32 /p%port60% /b%baud%

x32tobin /o %temp60%.x32 %temp60%.bin
del %temp60%.x32

60tosdf  /o %temp60%.bin
del %temp60%.bin

set port60=
set baud60=
 set temp60=
```

69_ML.BAT

Convert all HP 3569A data files that are transferred over the RS-232 port to PC-MATLAB Mat-file format. The default port is ''1'' (COM1). The default baud rate is ''9600''.

Syntax   69_ML.BAT [port] [baud]

```
@echo off

set port69=%1
set baud69=%2
set temp69=temp69
set dire69=tempsdf

if '%port69%'=='' set port69=1
if '%baud69%'=='' set baud69=9600

mkdir %dire69%
chdir %dire69%
echo y | del %dire69%

download /o %temp69%.x32 /p%port69% /b%baud%

x32tobin /o %temp69%.x32 %temp69%.bin
del %temp69%.x32

69tosdf  /o %temp69%.bin
del %temp69%.bin

for %%f in (*.dat) do sdftoml %%f /b /o >nul
copy *.mat ..
chdir ..

echo y | del %dire69%
rmdir %dire69%

set port69=
set baud69=
set temp69=
set dire69=
```

69_MX.BAT

Convert all HP 3569A data files that are transferred over the RS-232 port to MATRIXx Mat-file format. The default port is ''1'' (COM1). The default baud rate is ''9600''.

Syntax   69_MX.BAT [port] [baud]

```
@echo off

set port69=%1
set baud69=%2
set temp69=temp69
set dire69=tempsdf

 if '%port69%'=='' set port69=1
if '%baud69%'=='' set baud69=9600

mkdir %dire69%
chdir %dire69%
echo y | del %dire69%

download /o %temp69%.x32 /p%port69% /b%baud%

x32tobin /o %temp69%.x32 %temp69%.bin
del %temp69%.x32

69tosdf  /o %temp69%.bin
del %temp69%.bin

for %%f in (*.dat) do sdftomx %%f /b /o >nul
copy *.mat ..
chdir ..

echo y | del %dire69%
rmdir %dire69%

set port69=
set baud69=
set temp69=
set dire69=
```

69_PLOT.BAT

Plot all HP 3569A data files that are transferred over the RS-232 port. The default port is ''1'' (COM1).
The default baud rate is ''9600''.

Syntax   69_PLOT.BAT [port] [baud]

```
@echo off

set port69=%1
set baud69=%2
set temp69=temp69
set dire69=tempsdf

if '%port69%'=='' set port69=1
if '%baud69%'=='' set baud69=9600

mkdir %dire69%
chdir %dire69%
echo y | del %dire69%

download /o %temp69%.x32 /p%port69% /b%baud%

x32tobin /o %temp69%.x32 %temp69%.bin
del %temp69%.x32

69tosdf  /o %temp69%.bin
del %temp69%.bin

 for %%f in (*.dat) do viewdata %%f /t:%%f /p >nul
chdir ..

echo y | del %dire69%
rmdir %dire69%

set port69=
set baud69=
set temp69=
set dire69=
```

69_SDF.BAT

Convert all HP 3569A data files that are transferred over the RS-232 port to SDF. The default port is ''1'' (COM1). The default baud rate is ''9600''.

Syntax   69_SDF.BAT [port] [baud]

```
@echo off

set port69=%1
set baud69=%2
set temp69=temp69

if '%port69%'=='' set port69=1
if '%baud69%'=='' set baud69=9600

download /o %temp69%.x32 /p%port69% /b%baud%

x32tobin /o %temp69%.x32 %temp69%.bin
del %temp69%.x32

69tosdf  /o %temp69%.bin
del %temp69%.bin

set port69=
set baud69=
set temp69=
```

IB63_ML.BAT

Upload HP 3563A/62A active trace via HP-IB and convert it to PC-MATLAB Mat-file format.

Syntax   IB63_ML.BAT [filename]

```
@echo off

set file63=%1
if '%file63%'=='' set file63=HP3563

hpib63 %file63%.63 /o

63tosdf %file63%.63 %file63%.DAT /o
del %file63%.63

sdftoml %file63%.DAT /b /o >nul
del %file63%.DAT

 set file63=
```

IB63_MX.BAT

Upload HP 3563A/62A active trace via HP-IB and convert it to MATRIXx Mat-file format.

Syntax   IB63_MX.BAT [filename]

```
@echo off

set file63=%1
if '%file63%'=='' set file63=HP3563

hpib63 %file63%.63 /o

63tosdf %file63%.63 %file63%.DAT /o
del %file63%.63

sdftomx %file63%.DAT /b /o >nul
del %file63%.DAT

set file63=
```

IB63_PLT.BAT

Upload HP 3563A/62A active trace via HP-IB and plot it.

Syntax   IB63_PLT

```
@echo off

set file63=HP3563

hpib63 %file63%.63 /o

viewdata %file63%.63 /p
del %file63%.63

set file63=
```

IB63_SDF.BAT

Upload HP 3563A/62A active trace via HI-IB and convert it to SDF format.

Syntax
IB63_SDF [filename]

```
@echo off

set file63=%1
if '%file63%'=='' set file63=HP3563

hpib63 %file63%.63 /o

63tosdf %file63%.63 %file63%.DAT /o
del %file63%.63

set file63=
```

INDEX