Errata

Title & Document Type: 35665A Dynamic Signal Analyzer Concepts Guide

Manual Part Number: 35665-90028

Revision Date: September 1, 1991

HP References in this Manual

This manual may contain references to HP or Hewlett-Packard. Please note that Hewlett-Packard's former test and measurement, semiconductor products and chemical analysis businesses are now part of Agilent Technologies. We have made no changes to this manual copy. The HP XXXX referred to in this document is now the Agilent XXXX. For example, model number HP8648A is now model number Agilent 8648A.

About this Manual

We've added this manual to the Agilent website in an effort to help you support your product. This manual provides the best information we could find. It may be incomplete or contain dated information, and the scan quality may not be ideal. If we find a better copy in the future, we will add it to the Agilent website.

Support for Your Product

Agilent no longer sells or supports this product. You will find any other available product information on the Agilent Test & Measurement website:

www.tm.agilent.com

Search for the model number of this product, and the resulting product page will guide you to any available information. Our service centers may be able to perform calibration if no repair parts are needed, but no other support from Agilent is available.

Agilent Technologies

HP 35665A Dynamic Signal Analyzer

Concepts Guide

For Instruments with Firmwave Revision
A.01.00

HEWLETT
PACKARD

HP Part Number: 35665-90028
Microfiche Part Number: 35665-90228
Printed in U.S.A.

Print Date: September 1991

HEWLETT PACKARD

Safety Summary

The following general safety precautions must be observed during all phases of operation, service, and repair of this instrument. Failure to comply with these precautions or with specific warnings elsewhere in this manual violates safety standards of design, manufacture, and intended use of the instrument. Hewlett-Packard Company assumes no liability for the customer's failure to comply with these requirements. This is a Safety Class 1 instrument.

Ground The Instrument

To minimize shock hazard, the instrument chassis and cabinet must be connected to an electrical ground. The instrument is equipped with a three-conductor ac power cable. The power cable must either be plugged into an approved three-contact electrical outlet or used with a three-contact to two-contact adapter with the grounding wire (green) firmly connected to an electrical ground (safety ground) at the power outlet. The power jack and mating plug of the power cable meet International Electrotechnical Commission (IEC) safety standards.

Do Not Operate In An Explosive Atmosphere

Do not operate the instrument in the presence of flammable gases or fumes. Operation of any electrical instrument in such an environment constitutes a definite safety hazard.

Keep Away From Live Circuits

Operating personnel must not remove instrument covers. Component replacement and internal adjustments must be made by qualified maintenance personnel. Do not replace components with power cable connected. Under certain conditions, dangerous voltages may exist even with the power cable removed. To avoid injuries, always disconnect power and discharge circuits before touching them.

Do Not Service or Adjust Alone

Do not attempt internal service or adjustment unless another person, capable of rendering first aid and resuscitation, is present.

Do Not Substitute Parts or Modify Instrument

Because of the danger of introducing additional hazards, do not install substitute parts or perform any unauthorized modification to the instrument. Return the instrument to a Hewlett-Packard Sales and Service Office for service and repair to ensure the safety features are maintained.

Dangerous Procedure Warnings

Warnings, such as the example below, precede potentially dangerous procedures throughout this manual. Instructions contained in the warnings must be followed.

Warning **Dangerous voltages, capable of causing death, are present in this instrument. Use extreme caution when handling, testing, and adjusting.**

HEWLETT PACKARD

Safety Symbols

The following safety symbols are used throughout this manual and in the instrument. Familiarize yourself with each symbol and its meaning before operating this instrument.

General Definitions of Safety Symbols Used On Equipment or In Manuals.

	Instruction manual symbol: the product will be marked with this symbol when it is necessary for the user to refer to the instruction manual in order to protect against damage to the instrument.
	Indicates dangerous voltage (terminals fed from the interior by voltage exceeding 1000 volts must be so marked.)
OR	Protective conductor terminal. For protection against electrical shock in case of a fault. Used with field wiring terminals to indicate the terminal which must be connected to ground before operating equipment.
	Low-noise or noiseless, clean ground (earth) terminal. Used for a signal common, as well as providing protection against electrical shock in case of a fault. A terminal marked with this symbol must be connected to ground in the manner described in the installation (operating) manual, and before operating the equipment.
OR	Frame or chassis terminal. A connection to the frame (chassis) of the equipment which normally includes all exposed metal structures.
	Alternating current (power line).
	Direct current (power line).
	Alternating or direct current (power line).

Warning

The WARNING sign denotes a hazard. It calls attention to a procedure, practice, condition or the like, which if not correctly performed or adhered to, could result in injury or death to personnel.

Caution

The CAUTION sign denotes a hazard. It calls attention to an operating procedure, practice, condition or the like, which, if not correctly performed or adhered to, could result in damage to or destruction of part or all of the product.

Note

The NOTE sign denotes important information. It calls attention to procedure, practice, condition or the like, which is essential to highlight.

Guide to HP 35665A Documentation

If you are thinking about...	And you want to...	Then read...
♦ Unpack and install the HP 35665A	Install the HP 35665A Dynamic Signal Analyzer	*HP 35665A Installation and Verification Guide*
	Do operation verification or performance verification tests	*HP 35665A Installation and Verification Guide*
♦ Getting started	Make your first measurements with your new analyzer	*HP 35665A Quick Start Guide*
	Review measurement basics	*HP 35665A Concepts Guide*
	Learn what each key does	*HP 35665A Operator's Reference* or use the analyzer's [**Help**] key
♦ Making measurements	Learn how to make typical measurements with the HP 35665A	*HP 35665A Operator's Guide*
	Understand each of the analyzer's instrument modes	*HP 35665A Concepts Guide*
♦ Creating automated measurements	Learn the HP Instrument BASIC interface	*Using HP Instrument BASIC with the HP 35665A*
	Record keystrokes for a particular measurement	*HP 35665A Operator's Guide*
	Program with HP Instrument BASIC	*HP Instrument BASIC User's Handbook*
♦ Remote operation	Learn about the HP-IB	*HP–IB Programming with the HP 35665A*
	Learn how to program with HP-IB	*HP–IB Programming with the HP 35665A*
	Find specific HP-IB commands	*HP 35665A HP-IB Commands: Quick Reference*
♦ Servicing the analyzer	Adjust, troubleshoot, or repair the analyzer	*HP 35665A Service Guide*

Table of Contents

Part I - Overview

Part II - Instrument Modes

Chapter 16: Display Formats

Chapter 17: Scale Concepts

Part IV - Source/Capture Concepts

Chapter 18: Source Overview

1

Before You Begin

About this Book

This book is the *HP 35665A Concepts Guide*. This book provides a conceptual overview of the HP 35665A Dynamic Signal Analyzer and its essential features. It also contains backgound information to help you better understand and use the HP 35665A. Some of this information may be particularly useful if you haven't used an FFT analyzer before.

For the most part, this book does not contain task information or examples of typical measurements. For this information, see the *HP 35665A Operator's Guide*. Additionally, you should use the other books in the operating manual set. For more information, see "Where to find Additional Information" later in this chapter.

About the Analyzer

Introduction

The Hewlett-Packard 35665A Dynamic Signal Analyzer is a two-channel FFT spectrum/network analyzer with a frequency range that extends from nearly dc to just over 100 kHz. As such, the analyzer is a general-purpose design tool for measurement and evaluation of many electronic, electromechanical, and mechanical devices. The analyzer is also useful for noise and vibration analysis—and to make these measurements even easier, the analyzer has a built-in ICP current source for those accelerometers that require it.

Although the HP 35665A is primarily a frequency-domain analyzer, you can also use it to make time-domain and amplitude-domain measurements. To extend the analyzer's measurement capability, separate options are available for swept sine, rotating machinery, control systems, and acoustics applications.

You can also operate the analyzer remotely, via the HP-IB, to make automated measurements—a technique that's particularly useful for repetitive tasks (such as those encountered in production-line testing). However, you aren't restricted to remote operation for automated measurements—you can also use the analyzer's optional HP Instrument BASIC capability (and keystroke recording feature) to create automated measurement routines *without an external controller or computer.*

More than just a Spectrum/Network Analyzer

As we mentioned, the HP 35665A is more than just a two-channel spectrum/network analyzer. You could think of it as several instruments in a single case. When equipped with the full complement of options, the HP 35665A functions as a spectrum analyzer, network analyzer, acoustic sound level meter, acoustic intensity analyzer, vibration analyzer, digital oscilloscope, and amplitude-domain analyzer!

The standard HP 35665A has three *instrument modes*. To extend measurement capability, there are three other optional instrument modes. An instrument mode is, in effect, an individual "personality" that configures the analyzer to make specific types of measurements. For example, in FFT Analysis mode, the analyzer functions as a standard low-frequency FFT spectrum/network analyzer. In Octave Analysis mode, the analyzer functions as a real-time, parallel-filter acoustics analyzer.

The standard HP 35665A is equipped with the following instrument modes:

- FFT Analysis.
- Correlation Analysis.
- Histogram/Time.

Additionally, your HP 35665A may be equipped with the following optional instrument modes:

- Octave Analysis.
- Order Analysis.
- Swept Sine.

To learn more about instrument modes, see chapter 3, *"Getting Comfortable with the HP 35665A Dynamic Signal Analyzer."*

Other Options

In addition to the extended measurement capability available with the optional instrument modes, there are other options available for the HP 35665A Dynamic Signal Analyzer. These include:

- HP Instrument BASIC
- Curve Fit/Synthesis
- External Keyboard
- Additional Memory (two or six Megabytes)
- Arbitrary Waveform Source

Other Options

To see what options your HP 35665A analyzer has, press the [**System Utility**] hardkey and then press the [OPTIONS SETUP] softkey.

How to Use this Book

If You've Used an FFT Analyzer Before...

If you've used an FFT spectrum analyzer before, you should have no trouble making measurements with the HP 35665A. This instrument has many of the same features found in other Hewlett-Packard analyzers, such as the HP 35660A, HP 3561A, HP 3562A, HP SINE, and the HP 3566A/3567A.

To get comfortable with your new analyzer, continue reading the book you are using now—the *HP 35665A Concepts Guide*. If you want to step though a few quick measurement tasks, use the *HP 35665A Quick Start Guide*. Alternatively, you can proceed directly to the *HP 35665A Operator's Guide* and try several of the measurement tasks in that book.

If You Haven't Used an FFT Analyzer Before...

If you haven't used an FFT spectrum analyzer before, try one or two of the tasks in the *HP 35665A Quick Start Guide*. Then read the first part of the book you are reading now—the *HP 35665A Concepts Guide*. This contains essential background material to help you understand and use your new analyzer. In particular, be sure to review the chapters on "Spectrum Analyzer Basics" and "Getting Comfortable with the HP 35665A."

Afterwards, spend some time with the *HP 35665A Operator's Guide*. The sample measurements in that book are representative of many common measurements made with low-frequency spectrum analyzers. And since each task introduces the analyzer's features in a sequential, easy-to-understand fashion, you'll soon have the skills necessary to use the analyzer with total confidence. You will also find tasks that demonstrate measurements available with each of the analyzer's optional instrument modes.

Keep in mind that simply looking over the measurement tasks is not enough to really learn how to use the analyzer. If at all possible, gather the necessary equipment (outlined at the beginning of each task), set things up, and step through each measurement task. If you don't have a lot of time (or don't have the necessary equipment), make sure you have at least completed some of the tasks in the *HP 35665A Quick Start Guide.*

Firmware Revision Date

This book should be used with HP 35665A Dynamic Signal Analyzers having firmware version A.01.00. If your analyzer has a *significantly* different firmware revision, contact your local HP Sales/Service office to obtain a documentation set that matches your firmware revision date.

Firmware revisions are significant *only if the first two digits in the firmware revision date are changed.* For example, A.01.00 indicates a significant change from A.00.00. However, a change to A.00.01 from A.00.00 indicates very minor changes that do not affect the documentation set.

To check the firmware revision date of your instrument, press [**System Utility**] and then [S/N VERSION].

Need Assistance?

If you need assistance, contact your nearest Hewlett-Packard Sales and Service Office listed in the HP Catalog, or contact your nearest regional office listed at the back of this guide. If you are contacting Hewlett-Packard about a problem with your HP 35665A Dynamic Signal Analyzer, please provide the following information:

- Model number: HP 35665A
- Serial number:
- Firmware version:
- Options :
- Date the problem was first encountered:
- Circumstances in which the problem was encountered:
- Can you reproduce the problem?
- What effect does this problem have on you?

Notation Conventions

Hardkeys

Throughout this book, they are printed like this: [**Inst Mode**]. Hardkeys are front-panel buttons whose functions are always the same. Hardkeys have a label printed directly on the key itself.

Softkeys

Throughout this book, softkeys are printed like this: [FFT ANALYSIS]. Softkeys are keys whose functions change with the analyzer's current menu selection. A softkey's function is indicated by a video label to the left of the key (at the edge of the analyzer's screen).

Toggle Keys

Some keys toggle through different settings. Toggle softkeys have a highlighted word in their label that changes with each press of the softkey. Throughout this book, toggle softkeys are depicted as they *appear after you make the keypress*. For example, "toggle to [X-AXIS **LIN** LOG]" means to press [X-AXIS LIN/LOG] until the word LIN is highlighted.

There is only one toggle hardkey. This is the [**Pause-Cont**] hardkey.

Ghosted Softkeys

Occasionally, a softkey may be inactive—this occurs when a softkey is not appropriate for a particular measurement. When this happens, the analyzer "ghosts" the inactive softkey. For example, if you set the analyzer to one-channel mode, and then press [**Meas Data**], the [FREQUENCY RESPONSE] softkey will be ghosted. This is because frequency response measurements are only possible when the analyzer is in two-channel mode.

Where to find Additional Information

Using the [Help] key

The [**Help**] key on the analyzer's front panel provides fast, easy-to-read information about specific instrument controls and features. Using [**Help**] is particularly convenient when you need assistance and you don't have the *Operator's Reference* near at hand.

The [**Help**] key is also a good way to learn about the analyzer (or to refresh your memory if you don't use the analyzer very often). The help facility also has an index that lets you request information by key name or by topic.

The Quick Start Guide

Use the *HP 35665A Quick Start Guide* as an introduction to the HP 35665A. If you haven't read this book yet, you should probably do so. The Quick Start Guide is very short, but it's designed to get you comfortable with the analyzer by helping you make a sample measurement within fifteen minutes.

The Operator's Guide

The *HP 35665A Operator's Guide* contains tutorial-style tasks that demonstrate typical low-frequency spectrum/network analyzer measurements. Each of the analyzer's six instrument modes are demonstrated with at least one task (keep in mind that some of these modes are optional; your analyzer may not have all of them installed).

Additionally, the *HP 35665A Operator's Guide* contains tasks that go beyond basic measurements. For example, you'll find an introduction to limit testing and keystroke recording with the optional HP Instrument BASIC—features that make it easy to build automated measurement routines. You will also learn about plotting/printing measurement results and saving, recalling, and copying measurement data. Where appropriate, each of these chapters contains sample tasks to help you get comfortable with these features.

The Operator's Reference

For random-access information about the analyzer's controls and features, refer to the *HP 35665A Operator's Reference.* This is a dictionary-style reference that offers introductory tours of both front and rear panels, menu maps, and a short description of each hardkey and softkey.

Programmer's Reference

To help you operate the analyzer remotely via HP-IB, see *HP-IB Programming with the HP 35665A*. Here you'll find a conceptual overview of the HP-IB and how you can use it to control you instrument remotely. There is also a command reference that lists all HP-IB commands. This includes a description of each command, its proper syntax, and example statements. Additionally, there are sample programs to help you create your own HP-IB programs.

HP Instrument BASIC

To learn more about using HP Instrument BASIC (a subset of the HP BASIC programming language) with your new analyzer, see *Using HP Instrument BASIC with the HP 35665A*. This shows you how to record and develop programs for the HP 35665A. There are also sample programs to help you get started with HP Instrument BASIC.

For more global information about HP Instrument BASIC, see the *HP Instrument BASIC User's Handbook*. This is a reference for the HP Instrument BASIC language—a language used on the HP 35665A as well as other Hewlett-Packard instruments.

Installation and Verification Test Guide.

For specifications, installation instructions, and performance tests, see the *HP 35665A Installation and Verification Guide*.

Service Guide

For service information, see the *HP 35665A Service Guide*. This manual includes adjustments, replaceable parts, circuit descriptions, and troubleshooting.

Demonstration Disc

Consider ordering the *HP Dynamic Signals Demo Disc* (HP part number 35665-95900). This contains captured signals from microphones and vibration transducers for 72 different types of signals. These may be helpful as you learn to use the HP 35665A Dynamic Signal Analyzer—particularly if you are interested in making acoustics or vibration measurements.

To use the demonstration disc, you simply connect a standard audio Compact Disc player to the analyzer's input connectors. Each disk is shipped with documentation to explain the signals and to offer appropriate measurement suggestions. For more information, contact your local Hewlett-Packard Sales and Service Office.

Related Information.

For applications information, you can request copies of Hewlett-Packard application notes from your local HP Sales and Service Office. For a more general overview of spectrum/network measurements, see *Spectrum and Network Measurements* by Robert A. Witte (Prentice Hall, Englewood Cliffs, New Jersey, 1991).

2

Measurement Basics

The following chapter is a conceptual overview of spectrum analysis, with particular emphasis on FFT spectrum analyzers like the HP 35665A. As such, this chapter is designed for people who haven't used a spectrum analyzer. If you've already used a spectrum analyzer (and feel comfortable with the basics of spectrum analysis), proceed directly to chapter 3, "Getting Comfortable with the HP 35665A."

Spectrum Analyzer Basics

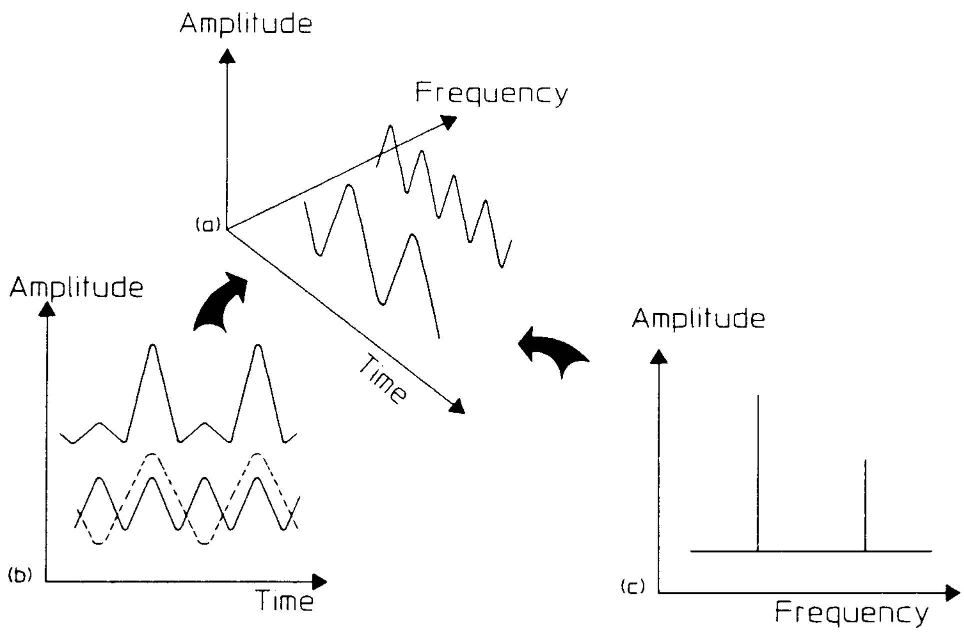
Introducing the Frequency Domain

If you haven't used a spectrum analyzer before, it's important to understand the difference between time-domain displays and frequency-domain displays.

A spectrum analyzer is an instrument used to characterize signals in the *frequency domain.* This is in contrast to the more traditional oscilloscope—an instrument that characterizes signals in the *time domain.*

Time-domain displays show a parameter (usually amplitude) versus *time.* This is the traditional way of looking at a signal. Oscilloscopes display signals in the time domain.

Frequency-domain displays show a parameter (again, usually amplitude) versus *frequency.* A spectrum analyzer takes an analog input signal—a time-domain signal—and converts it to the frequency domain (this conversion can be done in several ways; we'll talk about that a little later). The resulting spectrum measurement shows the energy of each frequency component at each point along the frequency spectrum.



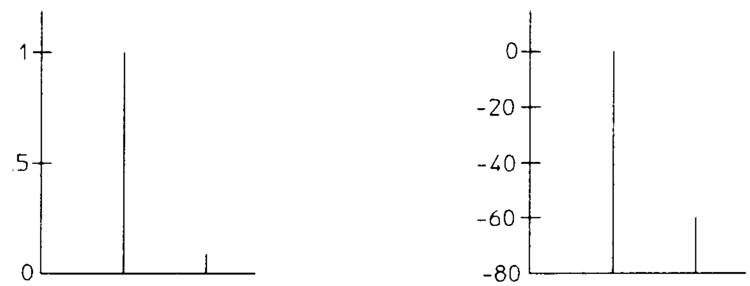
Notice the difference between the time-domain and frequency-domain displays of the same input signal.

Many signals not visible in the time domain (such as noise and distortion products) are clearly visible in the frequency domain. Because spectrum displays show frequency components distributed along the frequency axis, it's possible to view many different signals at the same time. This is why the spectrum analyzer is such a useful tool for looking at complex signals—it lets you easily measure (and compare) the frequency and amplitude of individual components. It also lets you track subtle changes over time—for example, the frequency drift of a signal source.

The Y-Axis (amplitude)

Time-domain measurements are usually viewed with a linear X-axis and a linear Y-axis (think of an oscilloscope). Frequency-domain measurements are sometimes viewed with a linear Y-axis and a linear X-axis, but usually must be viewed with a logarithmic Y-axis, since this is the only way to view very small signals with much larger signals.

Let's look at the spectrum of a sine wave. Because the amplitude of any harmonic is small relative to the fundamental frequency, it's nearly impossible to view a harmonic on the same display as the fundamental unless the Y-axis scale is logarithmic. That's why most measurements made with spectrum analyzers use a logarithmic amplitude scale—a scale based on decibels. And since the dB scale is by definition logarithmic, there's no need to use logarithmically-spaced graticule lines.



(a) Linear Amplitude Scale (b) Logarithmic Amplitude Scale

The X-Axis (frequency)

Sometimes it's convenient to use a logarithmic X-axis. Perhaps most familiar to you is the frequency response measurement. This is traditionally displayed with a log X-axis (frequency) versus a log Y-axis (relative magnitude).

But most measurements do not require a logarithmic frequency scale. In fact, when making spectrum measurements it's easier to characterize harmonics with a linear X-axis scale since harmonics that are multiples of the same fundamental will appear at evenly-spaced intervals.

There are some spectrum analyzer displays that use an X-axis other than frequency. For example, the HP 35665A Dynamic Signal Analyzer has an instrument mode called *order analysis*. This is used for rotating machinery applications. In this mode, the X-axis represents orders (one order represents a single revolution of a drive shaft; multiple orders represent harmonics of this speed). This is a much more convenient way to characterize the behavior of rotating machinery than plotting amplitude versus frequency.

What are the Different Types of Spectrum Analyzers?

There are two broad categories of spectrum analyzers: *swept-tuned analyzers* and *real-time analyzers*. Both swept-tuned analyzers and real-time analyzers have been around for many years. However, within the past decade or so, spectrum analyzers have become much more sophisticated. These newer spectrum analyzers use digital signal processing to provide additional measurement capablity—and let you interpret measurement results much more easily.

The increasing power of digital signal processing had produced more modern spectrum analyzers with many spectrum analyzers have become much more sophisticated—additional measuring capability measuring instruments, as they have incorporated powerful digital signal-processing technologies.

Both swept-tuned and real-time spectrum analyzers display amplitude versus frequency. How they process and display this information, however, varies with the specific type of analyzer. A real-time spectrum analyzer displays the energy at all frequency components simultaneously. A swept-tuned spectrum analyzer displays measurement results sequentially—in other words, not in "real time." This is because a swept-tuned analyzer, in effect, uses a narrow filter that is tuned across a range of frequencies to produce a spectrum display.

Swept-tuned analyzers have been the traditional choice for higher frequency applications—for example, 100 kHz and above. Real-time analyzers are generally used for lower frequencies—for example, audio-frequency and vibration measurements.

What is a Network Analyzer?

In its most rudimentary form, a *network analyzer* is a two-channel spectrum analyzer with a calibrated source—and one that is designed to compare the signal going into a device-under-test to the signal going out of the device-under-test. For example, if you place a filter between the network analyzer's source and one of the input channels, you can make measurements such as frequency response, gain, and phase flatness. Network analyzers are essential tools to evaluate the performance of electronic, electro-mechanical, and mechanical systems in the frequency domain. The HP 35665A Dynamic Signal Analyzer is both a spectrum analyzer *and* a network analyzer.

For additional overview of spectrum/network measurements, see *Spectrum and Network Measurements* by Robert A. Witte (Prentice Hall, Englewood Cliffs, New Jersey, 1991).

Swept-tuned Spectrum Analyzers

Swept-tuned spectrum analyzers are descended from radio receivers. So it should come as no surprise that swept-tuned analyzers are either *tuned-filter* analyzers (analogous to a TRF radio) or *superheterodyne* analyzers. In fact, in their simplest form, you could think of a swept-tuned spectrum analyzer as nothing more than a frequency-selective voltmeter with a frequency range that's tuned (swept) automatically.

Modern swept-tuned analyzers (superheterodyne analyzers, in particular) are precision devices that can make a wide variety of measurements. However, they are primarily used to measure steady-state signals since they can't evaluate all frequencies in a given span simultaneously. The ability to evaluate all frequencies simultaneously belongs exclusively to the *real-time* analyzer (more on this type of analyzer in a moment).

Tuned-filter Spectrum Analyzers

In a tuned-filter analyzer, a single bandpass filter is tuned across the frequency range of interest. The output of this filter is fed to a detector and then to an output device—either as a voltage on a meter or as a spectral line on a CRT screen.

Tuned-filter analyzers are rarely used nowadays for spectrum analysis, and for several reasons. For one thing, it's very difficult (and expensive) to make a tuned filter that maintains constant filter shape and bandwidth over a wide frequency range—the very same reason, in fact, why TRF radios became obsolete. It's even more difficult to make such a filter with variable bandwidths. Variable bandwidths are important in swept-tuned analyzers—using a wider bandwidth lessens the chance of missing a spectral component, while using a narrow bandwidth provides the resolution needed to isolate closely-spaced frequency components.

Superheterodyne Analyzers

A superheterodyne spectrum analyzer is similar to a high-performance superheterodyne radio receiver. This type of spectrum analyzer uses several local oscillators (and multiple mixing stages) to convert the input signal to several different intermediate frequencies. Like a good radio, multiple conversion is used to more easily reject image frequencies.

The superheterodyne analyzer uses a variable local oscillator to examine a range of frequencies (this eventually translates to a visual sweep between the start and stop frequencies of the spectrum you're examining). Superheterodyne analyzers also have a bank of fixed-tuned filters to provide various degrees of frequency resolution. Detected signals are then displayed as spectral lines on a CRT display.

Superheterodyne analyzers are widely used and have many advantages. They can have tremendous frequency ranges—for example, nearly dc to hundreds of MHz (or even hundreds of GHz). You can also set precise start and stop frequencies. And superheterodyne analyzers have wide amplitude ranges, excellent sensitivity and frequency stability, and high resolution of both frequency and amplitude.

Despite the high performance of modern superheterodyne analyzers, they still can't evaluate frequencies simultaneously. Thus, they are not *real-time* analyzers. And the sweep speed of a swept-tuned analyzer is always limited by the time required for its internal filters to settle.

Real-Time Spectrum Analyzers

Real-time analyzers are so named because they display an entire frequency spectrum simultaneously. Real-time analyzers are needed to measure transient and non-deterministic signals—signals that a swept-tuned analyzer cannot capture. Traditionally, real-time analyzers have been used in acoustics applications, where the relatively coarse resolution (one-third or full octave) is satisfactory—and in fact, desired—for many acoustics measurements.

Parallel-filter Analyzers

Another way to build a spectrum analyzer is to combine several bandpass filters, each with a different passband frequency. Each filter remains connected to the input at all times. This type of analyzer is called a *parallel-filter* analyzer.

If the center frequency of each filter is tuned to a particular part of the frequency spectrum (and each filter's response overlaps correctly), the combined response of the analyzer is sufficient to characterize a full range of frequencies. After an initial settling time (the time required for the "slowest" filter to settle), the parallel-filter analyzer can instantaneously detect and display all signal within the analyzer's measurement range.

The particular strength of such an analyzer is its measurement speed—this allows it to measure transient and non-deterministic signals. However, the frequency resolution of a parallel-filter analyzer is much coarser than a typical swept-tuned analyzer. This is because the resolution is determined by the width of the descriminating filters. To get fine resolution over a large frequency range, you would need many, many individual filters—thus increasing the cost and complexity of such an analyzer. This is why all but the simplest parallel-filter analyzers are expensive.

Typically, parallel-filter analyzers designed for audio-frequency applications use 32 filters. Each filter covers one-third of an octave. Many acoustics measurements are made exclusively with this type of analyzer.

FFT Analyzers

FFT spectrum analyzers use digital signal processing to sample the input signal and convert it to the frequency domain. This conversion is done using the *Fast Fourier Transform* (FFT). The FFT is an implementation of the Discrete Fourier Transform, the math algorithm used for transforming data from the time domain to the frequency domain. The HP 35665A Dynamic Signal Analyzer is an example of an FFT analyzer that can make real-time measurements.

FFT spectrum analyzers are powerful instruments, since their processing power can extract more information from an input signal than just the amplitude of individual frequency components. For example, FFT analyzers can measure both magnitude and phase, and can also switch easily between the time and frequency domains. This makes them ideal instruments for acoustic, modal vibration, and rotating machine analysis—and for evaluating many types of electronics devices as well.

If an FFT analyzer samples fast enough, all input data is evaluated and the analyzer will make a *real-time* measurement. When operating in real time, FFT analyzers can make the same measurements traditionally done with parallel-filter analyzers—and make these measurements, if desired, with far greater frequency resolution.

The only real disadvantage of FFT analyzers is their restricted frequency range—most FFT analyzers cannot make measurements above 100 kHz. The limiting factor is the speed of the analog-to-digital converter used to sample the analyzer's input signal. This is why swept-tuned superheterodyne analyzers are still used for RF and microwave measurements (though some newer-generation swept-tuned analyzers, such as the HP 3588A, can also make FFT measurements).

FFT Background

The Fast Fourier Transform

The Fourier transform integral converts data from the time domain into the frequency domain. However, this integral assumes the possibility of deriving a mathematical description of the waveform to be transformed—but real-world signals are complex and defy description by a simple equation. The Fast Fourier Transform (FFT) algorithm operates on sampled data, and provides time-to-frequency domain transformations without the need to derive the waveform equation.

FFT Properties

As with the swept-tuned analyzer, the input to the analyzer is a continuous analog voltage. The voltage might come directly from an electronic circuit (for example, a local oscillator) or through a transducer (for example, when measuring vibration). Whatever the source of the input signal, the FFT algorithm requires digital data. Therefore, the analyzer must convert the analog voltage in to a digital representation. So the first steps in building an FFT analyzer are to build a sampler and an analog-to-digital converter (ADC) in order to create the digitized stream of samples that feeds the FFT processor.

The FFT algorithm works on sampled data in a special way. Rather than acting on each data sample as it is converted by the ADC, the FFT waits until a number of sample (N) have been taken and transforms the complete block of data. The sampled data representing the time-domain waveform is typically called a time record of size-N samples.

But the FFT analyzer cannot compute a valid frequency-domain result until at least one time record is acquired—this is analogous to the initial settling time in a parallel-filter analyzer. After this initial time record is filled, the FFT analyzer is able to determine very rapid changes in the frequency domain. A typical size for N might be 1024 samples in one time record.

During the FFT process, the FFT algorithm transforms the N time domain samples into N/2 equally-spaced lines in the frequency domain. Each line contains both amplitude and phase information—this is why half as many lines are available in the frequency domain (actually, slightly less than half the number of lines are used, since some data is corrupted by anti-aliasing filters).

Frequency Resolution

The frequency resolution of an FFT analyzer is usually stated in number of lines. The most common offerings are 400 and 800 lines, although some analyzers—such as the HP 35665A— offer variable resolution. A 400-line FFT analyzer, set up to display a 4 kHz span, would have a frequency resolution of 4000 Hz per 400 lines—10 Hz per line. The length of the time record determines how long a given measurement will take and the maximum frequency that you can measure.

For example, an 800-line analyzer measuring a 1 kHz span requires a 0.8 second time record. A 3200-line analyzer measuring the same 1 kHz span requires a 3.2-second time record. This relationship is independent of processing speed. The smaller the span, the longer the time record required.

You can easily make calculations to determine values for four inter-related functions; maximum frequency, time-record length, frequency resolution, and frequency span:

- Maximum frequency = (N/2) x (1/time-record length), where N=number of samples
- Time record length = number of lines/frequency span
- Frequency resolution = 1/time-record length
- Frequency span = (frequency resolution x FFT lines)

What You Should Know about FFT Spectrum Analyzers

FFT Basics

As we mentioned, the Fast Fourier Transform (FFT) is a an implementation of the Discrete Fourier Transform, the math algorithm used for transforming data from the time domain to the frequency domain. Before an analyzer uses the FFT algorithm, it samples the input signal with an analog-to-digital converter (the Nyquist sampling theorem states that if samples are taken twice as fast as the highest frequency component in the signal, the signal can be reconstructed exactly). This transforms the continuous (analog) signal into a discrete (digital) signal.

Because the input signal is sampled, an *exact* representation of this signal is not available in either the time domain or the frequency domain. However, by spacing the samples closely, the analyzer provides an excellent approximation of the input signal.

Display Resolution and Frequency Span

FFT analyzers have a finite resolution, usually stated in number of "points," "lines," or "bins." Most FFT analyzers use the same number of display points regardless of frequency span. The HP 35665A, for example, has a default resolution of 400 lines (401 points, or bins)—and you can specify 800, 200, or 100 lines of display resolution. However, for a given display resolution, narrower spans have finer frequency resolution. This is because the same number of display points represents a smaller range of frequencies.

Full-span measurements let you view the entire frequency spectrum on one display. With the HP 35665A, the spectrum can extend from 0 Hz to 102.4 Hz for one-channel measurements up to 51.2 kHz for two-channel measurements. This provides a frequency resolution, at full-span, of 256 Hz and 128 Hz respectively.

Alternatively, you may wish to view smaller slices of the frequency spectrum. You can select any number of different spans and position these spans where you want by specifying their start or center frequencies. You can also select different spans by specifiying a center frequency and a span width. This process of viewing smaller spans is sometimes called "zooming" or "band-selectable analysis."

Measurements with spans that start at 0 Hz are often called *baseband* measurements. Those with spans that start at frequencies other than 0 Hz are often called *zoomed* measurements.

Zero Response and DC Measurements

What is Zero Response?

When viewing frequency spans that start at 0 Hz (or very close to 0 Hz), a spectral line is usually visible at the extreme left of a spectrum analyzer's display. This is zero response. In an FFT spectrum analyzer, zero response is caused by residual dc that originates in the analyzer's own input amplifiers. Zero response gives the illusion of a dc offset, even if the input signal has no dc component—and this occurs even if the analyzer has an ac-coupled input.

In the HP 35665 Dynamic Signal Analyzer, some degree of zero response is always present in the 0 Hz bin (sometimes called the dc bin). The residual dc that causes this offset may also leak into the first several bins as well. If you don't see any zero response on the analyzer's display, simply start the frequency span several bins above 0 Hz.

Can Spectrum Analyzers Measure DC?

Most spectrum analyzers are not intended to measure dc. However, analyzers such as the HP 35665A can measure very low frequencies—as low as 244 μHz for two-channel measurements and 488 μHz for one-channel measurements. The HP 35665A can, in fact, measure dc, but not without including a dc offset of its own that can contribute to (or obscure) a dc offset in the input signal. As we mentioned, this internal offset is caused by residual dc that originates in the analyzer's input amplifiers. Thus, measurement performance at dc is not specified.

The Time Record

A time record is the amount of time-domain data the analyzer needs to perform one FFT operation. The time record and its FFT are the building blocks the analyzer needs for all subsequent measurements.

For example, with the default display resolution of 400 lines, the HP 35665A Dynamic Signal Analyzer takes 1024 samples of time data to produce 512 points of frequency domain data. The analyzer usually displays the first 401 points of this data and discards the rest (this accommodates the anti-aliasing filters, but that's beyond the scope of our current discussion).

The relationship between a time record and the frequency data is relatively straightforward. If a signal component completes one cycle within the time record, it will show up in the first frequency "bin" (the first point on the analyzer's display). If a component takes two cycles to complete, it will show up in the second bin. And so forth.

So if a time record is one second long (and you start at 0 Hz), then the period of the signal for the first bin is also one second. And its frequency (1/period) is 1 Hz. Since there are 401 bins displayed, the span will be 400 Hz. The effective sampling frequency is simply 1024 divided by the length of the time record.

Why is a Time Record Needed?

Essentially, the time record is a block of time-domain sample points. Now since the actual Fourier Transform does not have explicit time or frequency references (it simply operates on a sequential collection of points), FFT analyzers must assign arbitrary start and finish times for data to be transformed. These blocks of input data are called time records.

Measurement Speed versus Time Record Length

The size of a time record is inversely proportional to the frequency span. For smaller spans, an FFT analyzer needs a longer time record and therefore takes longer to make a measurement. For larger spans, an FFT analyzer needs a shorter time record and can therefore make a measurement much faster.

The nearby table shows the time required for the HP 35665A Dynamic Signal Analyzer to make measurements using different frequency spans. The different times required will become noticeable as you start making measurements.

The characteristic time/span relationship for the HP 35665A is a natural part of the FFT process and is common to *all* FFT analyzers, not just the HP 35665A. Swept-tuned analyzers, by the way, have similar limitations (and are, in fact, *much* slower than FFT analyzers for comparable measurements).

Measurement speed versus time record length (400-line resolution)

Frequency Span (in Hz)	Time Record Length (in seconds)*	Resolution (in Hz)**
102400 †	0.00390625	256
51200	0.0078125	128
25600	0.015625	64
12800	0.03125	32
6400	0.0625	16
3200	0.125	8
1600	0.25	4
800	0.5	2
400	1	1
200	2	0.5
100	4	0.25
50	8	0.125
25	16	0.0625
12.5	32	0.03215
6.25	64	0.015625
3.125	128	0.0078125
1.5625	256	0.00390625
0.78125	512	0.001953125
0.390625	1024	0.0009765625
0.1953125	2048	0.00048828125
0.09765625 ‡	4096	0.00024414625

* These values are for the analyzer's default display resolution (400 lines). If you are using the 800-line display resolution, multiply the values in this column by 2. For 200-line resolution, divide by 2. For 100-line resolution, divide by 4.

** Frequency equals (current span)/400. If you are using the 800-line display resolution, divide the values in this column by 2. For 200-line resolution, multiply by 2. For 100-line resolution, multiply by 4.

† 1-Channel mode only.

‡ 2-Channel mode only.

Windowing

A *window* is a time-domain weighting function applied to the input signal. A window is a filter used to remove signals that are not periodic (and therefore spurious) within the input time record. Depending on the window, the analyzer attenuates certain parts of the input time record, to prevent *leakage*—a smearing of energy across the frequency spectrum—caused by transforming signals that are not periodic within the time record.

FFT analyzers usually have several window types available. Each window offers particular advantages. Because each window type produces different measurement results (just *how* different depends on the characteristics of the input signal and how you trigger on it), you should carefully select a window type appropriate for the measurement you're trying to make.

Windowing is a concept basic to understanding FFT spectrum analyzers. To learn more, see *Hewlett-Packard Application Note 243* (available from your local HP Sales/Service Office).

The HP 35665A Dynamic Signal Analyzer offers the following window types:

- Hanning
- Flat Top
- Uniform
- Force
- Exponential

To learn more about windowing, see chapter 7, "Windowing."

3

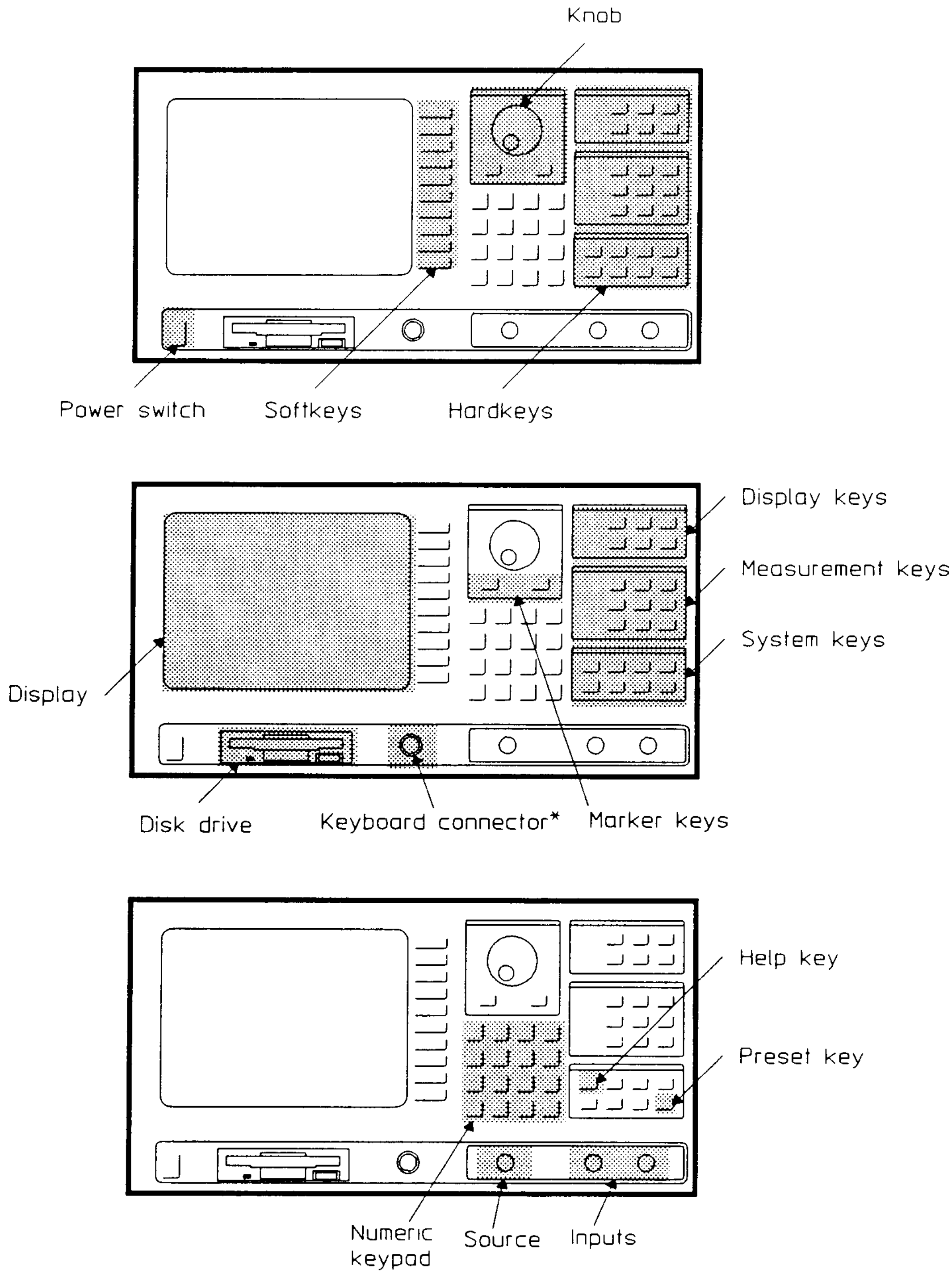
Getting Comfortable with the HP 35665A Dynamic Signal Analyzer

This is a very important chapter. It provides a brief overview of the HP 35665A Dynamic Signal Analyzer. After reading this chapter, you should understand each of the analyzer's six instrument modes—and the wide range of measurements available with each mode.

Before you start this chapter, make sure you've read chapter 1, "Before You Begin." If you haven't, go back and read it now. In addition, if you haven't used an FFT spectrum analyzer before, go back and read chapter 2, "Spectrum Analyzer Basics."

The HP 35665A at a Glance

For a detailed overview of the analyzer's front panel, see the HP 35665A Operator's Reference.



* may be rear-panel mounted on some instruments

Before You Get Comfortable

Hardkeys and Softkeys

If you've read "Notation Conventions" in chapter 1, you should already know the difference between hardkeys and softkeys.

Hardkeys are front-panel buttons whose functions are always the same. Hardkeys have a label printed directly on the key itself. Throughout this book, they are printed like this: [**Hardkey**].

Softkeys are keys whose functions change with the analyzer's current menu selection. A softkey's function is indicated by a video label to the left of the key (on the edge of the analyzer's screen). Throughout this book, softkeys are printed like this: [SOFTKEY].

Some softkeys toggle through different settings. Toggle softkeys have a highlighted bar in their label—this bar moves to a different part of the label with each press of the softkey. Throughout this book, toggle softkeys are depicted as they *appear after you make the keypress*. For example, "toggle to [X-AXIS **LIN**LOG]" means to press [X-AXIS LIN/LOG] until the word LIN is highlighted.

Occasionally, a softkey may be inactive—this occurs when a softkey is not appropriate for a particular measurement. When this happens, the analyzer "ghosts" the inactive softkey. For example, if you set the analyzer to one-channel mode, and then press [**Meas Data**], the [FREQUENCY RESPONSE] softkey will be ghosted. This is because frequency response measurements are only possible when the analyzer is in two-channel mode.

The Knob

The *knob* is an RPG (rotary pulse generator) that controls two things—movement of the on-screen marker and continuous entry of numeric values. Usually, the knob simply moves the marker (see chapter 14 to learn more about the marker). But after pressing a softkey that allows a numeric entry, the knob becomes dedicated to numeric entry. Turn the knob to the right (clockwise) and the analyzer steps through larger numeric entries. Turn to the left (counterclockwise) and the analyzer steps through smaller entries.

When numeric entry is active, an entry box appears at the top of the screen with the currently-selected numeric value. This box remains on screen for several seconds to give you a chance to enter a numeric value. After using the knob (or, alternatively, the numeric entry keypad) this box soon disappears and the knob returns to marker movement.

Alpha-Shift Hardkeys

It is occasionally necessary to specify alpha characters—for example, when entering a trace title or when saving or recalling a specific file. During these times, the analyzer automatically shifts certain front-panel keys to alpha entry keys (note the alpha characters engraved on the front panel below these hardkeys). When it's no longer necessary to enter alpha characters, the analyzer automatically returns these hardkeys to their normal functions.

Using the Optional Keyboard...

The HP 35665A analyzer has a connector that lets you attach an optional alphanumeric keyboard. You can use the keyboard to perform the same functions as you would using the front-panel alpha keys—for example, when specifying filenames or when entering a trace title. And using the keyboard makes it much easier to edit HP Instrument BASIC programs.

It's important to know that the keyboard remains active *even when the analyzer is not in alpha entry mode*. This means that you can operate the analyzer using the external keyboard rather than the front panel. Pressing the appropriate keyboard key does the same thing as pressing a hardkey or a softkey on the analyzer's front panel. To learn how the external keyboard maps to the anlalyzer's front-panel keys, see the *HP 35665A Operator's Reference*.

Caution

Use only the approved keyboard for this product. Hewlett-Packard does not warrant damage or performance loss caused by a non-HP approved keyboard. Currently, approved Hewlett-Packard keyboards are as follows:

- U.S. ASCII (C1405A #ABA)
- U.K. English (C1405A #ABU)
- German (C1405A #ABD)
- French (C1405A #ABF)
- Italian (C1405A #ABZ)
- Spanish (C1405A #ABE)
- Swedish/Finnish (C1405A #ABS)

Note

In addition to the U.S. English keyboard, the HP 35665A Dynamic Signal Analyzer supports French, German, Italian, Spanish, U.K./English, and Swedish/Finnish keyboards. To configure your analyzer for a keyboard other than U.S. English, press [**System Utility**] [KEYBOARD SETUP]. Then press the appropriate softkey to select the language.

Configuring your analyzer to use a different keyboard only ensures that the analyzer recognizes the proper keys for that particular keyboard. Configuring your analyzer to use another keyboard *does not* localize the on-screen annotation or the analyzer's online HELP facility.

Choosing an Instrument Mode

As we mentioned in chapter 1, the HP 35665A is more than just a two-channel spectrum/network analyzer. Instead, it's really like having several instruments in a single case. When equipped with the full complement of options, the HP 35665A functions as a spectrum analyzer, network analyzer, acoustic sound level meter, acoustic intensity analyzer, vibration analyzer, audio-frequency oscilloscope, and amplitude-domain analyzer!

To make it easier to use each of these different "instruments," you can press the [**Inst Mode**] hardkey. Then press the appropriate softkey to select an appropriate instrument mode. The standard HP 35665A is equipped with the following instrument modes:

- FFT Analysis.
- Correlation Analysis.
- Histogram/Time.

Additionally, your HP 35665A may be equipped with the following optional instrument modes:

- Octave Analysis.
- Order Analysis.
- Swept Sine.

Keep in mind that some measurement features are unavailable with certain instrument modes. As you use the analyzer, you will notice that some of its softkey menus change depending on the instrument mode you've selected. For example, *waveform capture* analysis (the ability to capture data in the analyzer's time capture buffer, for later analysis) is not available with the Swept Sine mode.

To learn more about the measurements you can make with each instrument mode, see the chapters in *Part II—Instrument Modes.*

Choosing Measurement Data and a Trace Coordinate

Once you've selected an instrument mode, you can press [**Meas Data**] to see the measurement data available with that particular instrument mode. Then, press [**Trace Coord**] to select an appropriate way to look at this measurement data. To see a list of the measurement data available with each instrument mode, see "Overview of Available Measurement Data" later in this chapter.

The dB magnitude trace coordinate is the most common way to view measurement data. However, other trace coordinates are also useful and can reveal information not visible from the dB magnitude display. Here is a list of the available trace coordinates (this list is the same regardless of the instrument mode you've selected):

- Linear Magnitude
- Logarithmic Magnitude
- dB Magnitude
- Phase
- Unwrapped Phase
- Real Part
- Imaginary Part
- Nyquist Diagram
- Linear or Logarithmic X-axis

These trace coordinates are available with most combinations of instrument mode and measurement data. There are, however, some cases where certain trace coordinates don't apply to particular measurements—if this happens, the inappropriate softkey will be "ghosted" (see "Hardkeys and Softkeys" earlier in this chapter). For example, when you select the FFT analysis instrument mode and then select an orbit measurement, all trace coordinates except *real part* are ghosted.

Instrument Mode? Measurement Data? Trace Coordinates?

Instrument Modes

When you specify an instrument mode, you are asking the analyzer to acquire input data and process it in a certain way. For example, with the FFT Analysis mode, the analyzer takes the FFT of a single time record and converts it to the most basic spectrum measurement available—the linear spectrum. For Swept Sine mode, the analyzer uses its sine source to sweep through a desired frequency range.

Look at the front panel. Notice how the [**Inst Mode**] is in the **MEASUREMENT GROUP**. This is because anything you change from this group of keys affects the way the analyzer collects input data. If you change something with these keys (such as the instrument mode, or a different start or stop frequency), the analyzer must retake a measurement.

If you look at the analyzer's front panel, you can see that both the [**Meas Data**] and [**Trace Coord**] hardkeys are in the **DISPLAY** group. If you press a key in this group, all you are doing is selecting which type of data you want to display—you aren't changing the way the analyzer makes measurements.

Softkeys in the **MEASUREMENT** group, however, do affect the way the analyzer takes input data. For example, [**Inst Mode**] is in this group. So is [**Freq**], which you use to change the start and stop frequencies. So if you need to change a parameter under any of these menus (or select a different instrument mode), you will need to take new data.

Measurement Data

Suppose you've selected the FFT Analysis mode. Now you can select different types of measurement data—for example, the linear spectrum, the power spectrum, time data, or frequency response. No matter which you select, the analyzer does not have to aquire new data. When you select a type of measurement data, you are simply asking the analyzer to display a particular piece of the measurement data that's already been acquired.

Trace Coordinates

Once you select both an instrument mode and appropriate measurement data, choosing a trace coordinate simply tells the analyzer how you want to look at the selected measurement data. For example, if you are in the FFT Analysis mode and you are viewing power spectrum data, you could change the Y-axis scaling by selecting linear magnitude, logarithmic magnitude, or dB magnitude trace coordinates.

Available Measurement Data

Each of the six instrument modes has its own set of available measurement data. The following table shows the measurement data available with each instrument mode. For more detailed information about this measurement data, see the chapters in *Part II—Instrument Modes.*

Measurement data available with each instrument mode

	FFT analysis	Octave Analysis	Order Analysis	Swept Sine	Correlation Analysis	Histogram /Time
Power Spec CH1	yes	yes	yes			
Power Spec CH2	yes*	yes*	yes*			
Linear Spec CH1	yes			yes		
Linear Spec CH2	yes*			yes		
Time Channel 1	yes		yes	yes	yes	yes ‡
Time Channel 2	yes*		yes*	yes	yes*	yes ‡*
Windowed Time Chan 1	yes				yes	
Windowed Time Chan 2	yes*				yes *	
Frequency Response	yes*			yes		
Coherence	yes*†					
Cross Spectrum	yes*			yes		
Orbit	yes*		yes			
Math Function	yes	yes	yes	yes	yes	yes

* Available only when analyzer is in two-channel mode
† Available only when averaging is turned on
‡ In Histogram/Time mode, time data is called "unfiltered time" data; see chapter 13 for details

(table continued on next page)

Measurement data available with each instrument mode (continued)

	FFT analysis	Octave Analysis	Order Analysis	Swept Sine	Correlation Analysis	Histogram
Data Register	yes §	yes §	yes §	yes §	yes §	yes §
Waterfall Register	yes	yes	yes	yes	yes	yes
Capture CH1	yes	yes	yes		yes	yes
Capture CH2	yes *	yes *	yes *		yes *	yes *
Composite Power CH1			yes			
Composite Power CH2			yes *			
Order Track CH1			yes			
Order Track CH2			yes *			
RPM Profile			yes			
Normalized Variance CH1				yes		
Normalized Variance CH2				yes		
Auto Correlation CH1					yes	
Auto Correlation CH2					yes *	
Cross Correlation					yes *	
Histogram CH1						yes
Histogram CH2						yes *
PDF† CH1						yes
PDF† CH2						yes*
CDF‡ CH1						yes
CDF‡ CH2						yes *

* Available only when analyzer is in two-channel mode
† Probability Density Function
‡ Cumulative Density Function
§ Data registers are unique in that they allow you to transport data from one instrument mode to another

Available Trace Coordinates

As we mentioned earlier, it's important to understand that selecting measurement data and selecting *how to look at that data* are two different things. When you specify the measurement data (for example, LIN SPEC CHANNEL 1), you are selecting from a variety of available measurement results. When you specify the trace coordinate (for example, dB Magnitude), you are simply determining how you want the measurement results displayed.

Linear Magnitude

Linear magnitude coordinates show the magnitude of the current measurement data using a linear Y-axis scale.

Logarithmic Magnitude

Logarithmic magnitude coordinates show the magnitude of the current measurement data using a logarithmic Y-axis scale. *Traces shown with dB magnitude coordinates and logarithmic magnitude coordinates appear the same—the only difference is the the Y-axis is labeled differently.*

dB Magnitude

The dB Magnitude coordinates show the magnitude of the current measurement data using a dB scale. *Traces shown with dB magnitude coordinates and logarithmic magnitude coordinates appear the same—the only difference is the the Y-axis is labeled differently.*

Real Part

The real part trace type shows the real part of the measurement defined for the active trace. For time record waveforms that are complex (zoomed measurements), the real part is scaled to be one-half the value of the waveform shown for real value (non-zoomed) time records.

Imaginary Part

The imaginary part trace type shows the imaginary part of the measurement defined for the active trace. If there's no imaginary data, the waveform will be a flat line, showing zero magnitude.

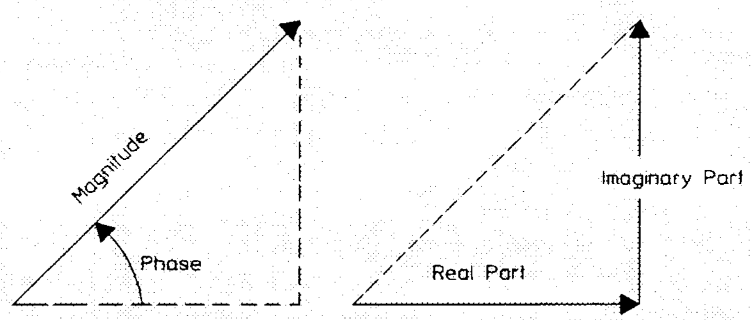
In FFT Analysis mode, the imaginary trace represents the imaginary part of the complex FFT data—for all data except time waveforms. For time waveforms, the imaginary trace represents the imaginary part of the Hilbert transform of the real part. For example, a 2 volt (peak) sine wave input in zoom mode will appear as a 1 volt (peak) sine in the real part trace, and as a 1 volt (peak) sine wave in the imaginary part trace (shifted 90 degress from the real part trace).

Real and Imaginary Parts

Once an input signal is transformed from the time domain to the frequency domain, there are two ways to express values for the components in each frequency bin. One choice is to specify the magnitude and phase of a component; the other is to show the real part and imaginary part of each component.

Polar Form (Magnitude and Phase). This is the most common way to characterize a frequency component. For example, when you select Log Mag and Phase as trace types, you are looking at magnitude and phase values for each frequency component. The magnitude represents the length of a vector and the phase is the angle of the vector.

Rectangular Form (Real and Imaginary parts). This is a less common way to characterize a frequency component. Still, it may be useful for some applications. For example, the imaginary part of the cross spectrum is used for acoustic intensity measurements. Real and imaginary parts can also correspond directly to the resistive and reactive components of an electrical network impedance.



The same frequency component can be expressed as a polar coordinate or a rectangular coordinate.

Phase

This coordinate shows *wrapped* phase versus frequency. *Wrapped* phase means that all phase is displayed within a range of –180 degrees and +180 degrees. If any point on the phase trace is outside this range, the analyzer adds or subtracts a multiple of 360 degrees to place this point within the –180 degree to +180 degree range.

For example, a phase of +400 degrees is displayed as +40 degrees (400 – 360 = 40). Conversely, a phase of –190 degrees is displayed as +170 degrees.

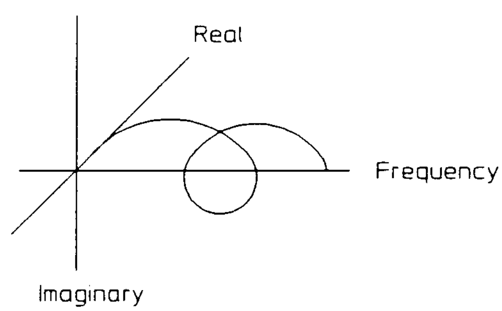
Unwrapped Phase

This coordinate shows *unwrapped* phase versus frequency. *Unwrapped* phase means that phase information (always referenced to the lowest measured frequency) is displayed. Phase information is not shifted to fall within the –180 degree to +180 degree range.

Nyquist Diagram

A *Nyquist Diagram* (often called a polar plot) shows the relationship between magnitude, phase, and frequency. A Nyquist diagram is unusual in that it lets you simultaneously view both the magnitude and the phase behavior of a device-under-test.

The general shape of a Nyquist diagram reflects the character of the test device. For example, Nyquist traces for bandpass filters often have a cardiod shape. Low-pass and high-pass filters have circular traces that spiral inward or outward.



Look at the above illustration. Imagine that you are looking down the end of the frequency axis—like looking down a tube or cylinder. All you can see of the frequency axis itself is a dot, since you are looking at it from the end. As you turn the marker knob, the marker readout shows both real and imaginary values for each frequency point.

Frequency response measurements displayed as Nyquist diagrams often show curved and spiraled traces. For a particular frequency response measurement, the amount that this trace curves indicates the amount of phase shift, as the wrapped phase accumulates. The distance from the edge of the Nyquist diagram indicates the magnitude. The "depth" of the spiral indicates relative frequency—but because you cannot tell where the trace is on the frequency axis (since you're viewing it from one end), you must rely on the marker readout to provide frequency information.

Y Units

Depending on the measurement data you've selected, you can choose different units for the vertical axis. You can also reference these to either dBV, dBm, dBSPL, and a user reference (to learn more about this, see the *HP 35665A Operator's Reference*).

The Y units are:

- Peak or rms amplitude.
- Degree or radian phase.
- Volts.
- Volts^2.
- Volts/rootHz (square root Power Spectral Density; voltage normalized to a 1 Hz bandwidth).
- Volts^2/Hz (Power Spectral Density; power normalized to a 1 Hz bandwidth).
- V^{2}s/Hz (Energy Spectral Density; energy normalized to a 1 Hz bandwidth).

Linear or Logarithmic X-axis

General Behavior

You can display measurement data on either a linear or a logarithmic X-axis. Keep in mind, however, that the analyzer's frequency resolution is determined exclusively by the width of the span and the display resolution (either 100, 200, 400, or 800 lines). So for identical span widths and display resolution, frequency resolution for both linear and log scales is identical. The logarithmic scale simply displays these points on a logarithmic X-axis. So when you display a trace on a logarithmic scale (with the exception of a log sweep in Swept Sine mode), there will be greater apparent resolution at the higher frequencies since the display points are closer together toward the right of the logarithmic display.

If you use a logarithmic X-axis for baseband measurements (spans that start at 0 Hz), the analyzer uses the first non-zero bin for the displayed start frequency, not the nominal value of 0 Hz. The analyzer does not show a value at 0 Hz since the log of 0 is negative infinity. For example, if you're looking at a 51.2 kHz span, the first frequency shown on the logarithmic scale is 128 Hz. If you needed to view components less than 128 Hz, you could change to a smaller span or move the center frequency downward.

Special Considerations with Swept Sine Mode

Unlike other instrument modes, the Swept Sine mode gives you the ability to specify either linear or logarithmic resolution—you select this by specifying either a linear or a logarithmic sweep. In addition, you can also vary the resolution of the sweep by specifying the number of points per sweep. To learn more about Swept Sine mode, see chapter 9.

4

Input Overview

Introduction

The HP 35665A has two input connectors on the front panel. Each input channel is single-ended and has an input impedance of 1 Megohm, shunted by a capacitance of less 100 pF. When the analyzer is in 1-channel mode, only channel 1 input is connected. When the analyzer is in 2-channel mode, both inputs are connected.

Here are some of the different ways to configure each input channel:

- Grounded/floating input.
- Coupling ac/dc.
- Anti-alias filter on/off.
- Autoranging on/off.
- A-weight filter on/off.
- ICP power supply on/off.

You can also specify engineering units from the input menu. Engineering units (EU) are useful when making measurements with calibrated transducers such as microphones or accelerometers. To learn more about engineering units, see "engineering units" later in this chapter.

In addition, the HP 35665A also has a *tachometer input* and an *external trigger* input. Both of these connectors are on the rear panel. To learn more about these inputs, see chapter 5.

Input Characteristics

Input Impedance

The analyzer's inputs are high impedance and are designed to provide measurements without loading down your device-under-test. Each input channel has a 1 Megohm input resistance shunted by less than 100 pF. If you need to load the output of your device-under-test with a specific impedance, be sure to place the proper load across the output of the test device.

Input Range

The analyzer has 40 input ranges, from –51 dBVrms to +27 dBVrms in 2 dB increments. You can set the input range manually, or you can use the analyzer's autoranging feature. If autoranging is on, the analyzer automatically selects a less-sensitive range if the signal exceeds the current input range. To make the best measurement possible, you should carefully consider the method you use to set the input range—setting the range manually or using autorange. To learn more about autoranging, see "Autoranging" later in this chapter.

Maximum Input Range (+27dBVrms) and its maximum values in equivalent units

dBVrms	dBV(peak)	Vrms	V(peak)
+27 dBVrms)	+30.01 dBV(peak)	+22.39 Vrms	+31.66 V(peak)

Minimum Input Range (–51dBVrms) and its maximum values in equivalent units

dBVrms	dBV(peak)	Vrms	V(peak)
–51 dBVrms	–47 dBV(peak)	+2.818 mVrms	+3.986 mV(peak)

Input Overload

There are the **Ov1** or **Ov2** status indicators, at the *top right-hand corner* of the analyzer's display. There is also an **OVLD** status message that may appear at the *bottom* of the analyzer's display. For a complete listing of all status indicators and status messages, see the *HP 35665A Operator's Reference.*

Ov1 or **Ov2** indicate if an input channel is currently overloaded. These indicators—like the analyzer's other status indicators—are normally ghosted. When an overload condition occurs, the **OV1** or **Ov2** indicators change from a ghosted state to non-ghosted state (full brightness).

The **OVLD** status message appears if an input channel is overloaded and the analyzer is unable to change to a less-sensitive range. This occurs if you exceed the analyzer's maximum input range (+27 dBv). If you turn off autorange, the **OVLD** message appears anytime you exceed the current input range.

The analyzer's response to an overload condition varies. If autoranging is on, an overload condition simply causes the analyzer to change to a less-sensitive input range—unless the maximum input range is already selected. If you are making an averaged measurement and an overload occurs, the analyzer begins a new series of averages.

In some cases, the **OVLD** message remains on the display even when there is no longer an overload condition. This lets you know that an overload condition has affected a measurement in progress. Consider this example: you are making an averaged measurement using a fixed input range. At some point during the measurement, an overload occurs. Because the analyzer does not autorange and restart the measurement (because autoranging is off), the **OVLD** message remains to let you know that an overload conditions has corrupted the averaged data.

Caution

Although the analyzer's input has protection circuitry, signals greater than the following (referenced to ground) may damage the analyzer:

- ± 42 volts (peak) applied to the center conductor of the input connector.
- ± 4 volts (peak) applied to the shell of the input connector (in float mode).

Note

Input overload detection is sensitive to broadband signals—this means that the analyzer senses overload conditions even if the over-range frequency component is outside the analyzer's current frequency span. Thus the **Ovl** or **Ov2** status indicators or the **OVLD** status message may appear even if you're viewing a frequency span with no over-range components.

Overload Reject

You should also know that the analyzer has an *overload reject* feature. You can turn on overload reject by pressing [**Avg**] and then toggling to [OVLD REJ **ON** OFF]. When overload reject is on, the analyzer does not make a measurement while the input channel is overloaded.

Overload reject operates with both single measurements and averaged measurements. With single measurements, the analyzer rejects the current measurement, performs an autorange procedure, and then begins another measurement. With averaged measurements, the analyzer stops the average count, the measurement process is suspended during an overload condition and resumes only if the overload condition is removed (since the analyzer rejects overloaded time records) autoranges if an overload occurs.

Autoranging

When autoranging is on, the analyzer continuously monitors the amplitude of the input signals and, if necessary, automatically changes the input range. If the input signal increases enough to exceed the current input range, the analyzer changes to a less-sensitive input range. When autoranging occurs, the analyzer displays the **Autorange in progess** message. If you are making an averaged measurement and autoranging occurs, the analyzer begins a new series of averages.

Autoranging for the HP 35665A is an "autorange up" feature. This means that when you start a measurement, the analyzer sets the input to the most sensitive range, and automatically steps through less-sensitive input ranges until the input channel is no longer overloaded. If the input signal amplitude decreases, the analyzer *does not* change to a different range—the input range remains at the setting the analyzer found appropriate at the beginning of the measurement.

Note

Autoranging for the HP 35665A is an "autorange up" feature. To preserve dynamic range and ensure the best measurement possible, you may want to initiate an autorange procedure if the input signal falls below 2 dB of the top of the current input range. To autorange, simply press [**Input**] and then press either [CH1 AUTORANGE] or [CH2 AUTORANGE]. To learn more about monitoring the input signal for optimal input range, see the "Monitoring Input Range" sidebar elsewhere in this chapter.

Fixed Input Range

You can set the input range manually when you want to maintain a specific input range setting. Ideally, the signal peak should fall in the upper half of the currently-selected input range.

If you set the input range too low (more sensitive than necessary), the analyzer's input circuitry will introduce distortion into the measurement. But if you set the input range too high (less sensitive than necessary), the resulting loss of dynamic range will introduce additional noise—in some cases, the increase in the noise floor may even obscure low-level frequency components.

If you use a fixed input range, there is approximately 2 dB of headroom at the top of each range before the input channel distorts. So if you are using a fixed input range, an overload condition does not occur unless you exceed this level. This is important to know, particularly for rotating machinery measurements where transient signals often exceed the desired input range.

Input Channel Configuration

Grounded/Floating Input

You can select either a grounded or a floating configuration for each input channel. Both grounded and floating inputs have an input resistance of 1 Megohm, shunted by a capacitance of less than 100 pF.

You should also know that:

- In the grounded configuration, there is a 55 ohm resistance from the shell of the input connector to the analyzer's chassis ground.
- In the floating configuration, there is a 1 Megohm resistance from the shell of the input connector to the analyzer's chassis ground. This configuration is actually a "pseudo floating" input since the input connector is not completely isolated from the chassis ground.
- Both the floating and the grounded inputs have a 1 Megohm resistance from the center conductor to the shell of the input connector.

Coupling AC/DC

You can select either ac or dc coupling for each input channel. With ac coupling, the input signal rolls off 3 dB at 1 Hz. Be sure to use dc coupling if you need to measure low-frequency signal components.

You should also use ac coupling if you've turned on the ICP supply for this input channel. See "ICP Power Supply On/Off" later in this chapter.

Anti-Alias Filter ON/OFF

FFT analyzers use anti-aliasing filters to provide accurate transformation of a signal from the time domain to the frequency domain. For most measurements, the analyzer's anti-aliasing filter should remain on. However, if you're looking at time domain measurement data, you can turn off the anti-aliasing filter if ringing or other distortion is a problem. This can be a problem with square waves, transients, or pulses that may have substantial high-frequency components.

In addition, there are some instrument modes that do not need an anti-aliasing filter since they do not transform data from the time domain to the frequency domain. Because Correlation Analysis mode and Histogram/Time mode are time-domain measurements, the anti-aliasing filters are not necessary.

You may want to turn off the anti-aliasing filters if you are using the time capture feature. To learn more about time capture, see chapter 19, "Capture Concepts."

A-Weight Filter ON/OFF

The A-weight function is often used for acoustics measurements and for some measurements of audio-frequency devices. The A-weight filter approximates the frequency response of the human ear—this is important because the human ear is more sensitive to certain frequencies than others.

The A-weighting filter is particularly useful if you are using the analyzer's Octave Analysis mode (Option 1D1). However, the A-weight filter is part of the input setup menu so you can use it with any of the analyzer's instrument modes.

Note

A-weighting is also available as a math function, as are B-weighting and C-weighting. These are software functions and are applied after data is acquired. To learn more about math functions, see chapter 20.

ICP Power Supply ON/OFF

For vibration measurements and modal analysis (impact testing), you can use the analyzer's built-in ICP power supply when using ICP (Integrated Circuit Piezoelectric) accelerometers. When you turn on the ICP power supply, a 4 mA current source appears at the center conductor of the appropriate input channel connector. The nominal voltage for the ICP supply is 24 Vdc (open circuit).

Note

In general, you should use ac-coupling for the input channel if the ICP supply is on. The ICP voltage is, in effect, a dc bias applied to the input connector. Turning on ac coupling blocks this dc bias. If you turn on dc coupling, the ICP supply forces the analyzer to change to a less-sensitive input range (if autoranging is on), thus degrading the signal-to-noise ratio for your measurement. If you've turned autoranging off, using a dc-coupled input with the ICP supply can overload the analyzer's front end and cause distortion.

Engineering Units

An "engineering unit" (EU) is an arbitrary unit to which you can assign any voltage value. You set up engineering units from the [Input] menu using the [CHANNEL 1 SETUP] or [CHANNEL 2 SETUP] softkeys.

When you use engineering units, you can get the analyzer to show results in non-voltage units—for example, mils, inches/second, or g's. This is useful when making rotating machinery or vibration measurements.

You can also use engineering units to make acoustics measurements with a calibrated microphone. To learn more about using engineering units for acoustics measurements, see the sample acoustics measurement in the *HP 35665A Operator's Guide.*

Engineering units are useful because they let you effectively convert a transducer's output voltage to any numerical value. They also let you assign an arbitrary label to an engineering unit that you've created.

To use engineering units, you enter an *engineering unit multiplier* using the [ENG UNIT MULTIPLIER] softkey. Then you use the [ENG UNIT LABEL] softkey to assign a label that describes the engineering unit you've created.

Note

When using engineering units, the transducer you are using must be a linear device. Engineering units are valid only when the relationship between the engineering unit and the transducer's output voltage is linear. For example, you can specify that the transducer output is 10 mV per engineering unit.

Also, an offset is not allowed. For example, you cannot add a value to form an engineering unit—you can only use a multiplier.

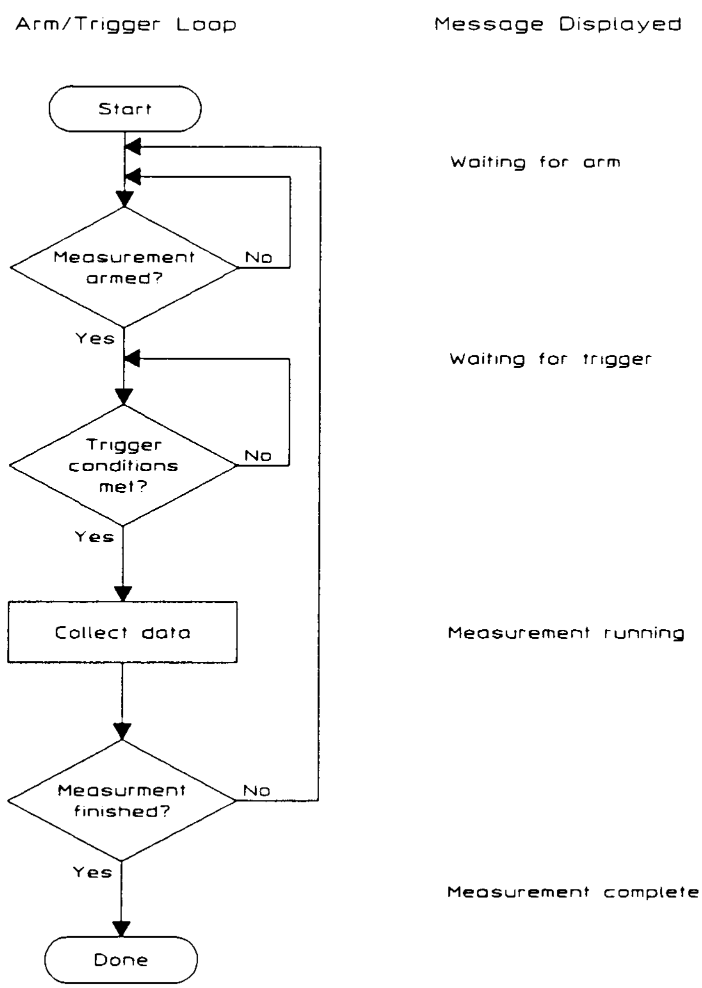
5

Triggering and Arming

Introduction

Starting a measurement is a three-state process. The first stage is pressing the [**Start**] key, the second stage is arming the measurement, and the third stage is triggering.

Arming merely enables the analyzer to respond to a trigger. Triggering tells the analyzer to begin collecting data. Both arming and triggering can be automatic or subject to specified conditions. If you specify automatic arming and free run triggering, the analyzer begins collecting data as soon as you press the [**Start**] key.



As with many other measuring instruments (for example, oscilloscopes and logic analyzers), triggering can be very important when making measurements with a spectrum/network analyzer. The HP 35665A has several different types of triggering. You can select a particular trigger to ensure that the analyzer samples input data at an appropriate time—for example, to synchronize a measurement with an external device.

Triggering allows you to display a stable waveform with a predictable and repeatable starting point. This is most obvious with time domain displays. For example, if you select free run triggering and display a 1 kHz sine wave in the time domain, the signal may appear to jump around on the screen. This is because the trigger is occurring at different points in the cycle. If you select channel 1 triggering, the trigger always occurs at the same point in the cycle, and the display is stable.

Arming

Arming merely enables the analyzer to respond to triggering. The analyzer performs only one measurement each time it is armed. The most common reason to use arming is to prevent a measurement from occurring until certain conditions are met.

The analyzer provides four kinds of arming:

- Automatic Arm. The analyzer is always armed and responds to trigger events immediately.
- Manual Arm. You must press the [ARM] softkey to re-arm the analyzer after each measurement.
- rpm Step Arm. The analyzer re-arms itself at specified rpm intervals of the tachometer input signal.
- Time Step Arm. The analyzer re-arms itself at specified time intervals.

The following table lists the types of arming available with each instrument mode. For detailed information on specific arming modes, see the *HP 35665A Operator's Reference* or the analyer's online help.

Arm type	Instrument Mode					
	FFT analysis	Octave Analysis	Order Analysis	Swept Sine	Correlation Analysis	Histogram/ Time
Automatic arming	yes	yes	yes		yes	yes
Manual arming	yes				yes	yes
rpm Step arming	yes	yes	yes		yes	yes
Time Step arming	yes	yes	yes		yes	yes

Triggering

The analyzer begins collecting data when the specified trigger conditions are met. The analyzer provides six types of triggering:

- Free Run Trigger. The analyzer collects data continuously, without waiting for a trigger signal.
- External Trigger. The analyzer triggers when the specified trigger conditions are met at the external trigger input on the rear panel.
- Channel 1 Trigger. The analyzer triggers when the specified trigger conditions are met at the channel 1 input connector.
- Channel 2 Trigger. The analyzer triggers when the specified trigger conditions are met at the channel 2 input connector.
- Source Trigger. The analyzer triggers from the internal source.
- HP-IB Trigger. The analyzer triggers when it receives an appropriate HP-IB command.

The following table lists the trigger types available for each instrument mode. For detailed information on specific trigger types, see the *HP 35665A Operator's Reference* or the analyzer's online help.

Trigger Type	Instrument Mode					
	FFT analysis	Octave Analysis	Order Analysis	Swept Sine	Correlation Analysis	Histogram/ Time
Free Run	yes	yes	yes		yes	yes
External	yes	yes	yes		yes	yes
Channel 1	yes				yes	yes
Channel 2	yes *				yes *	yes *
Source	yes					yes
HP-IB	yes	yes			yes	yes

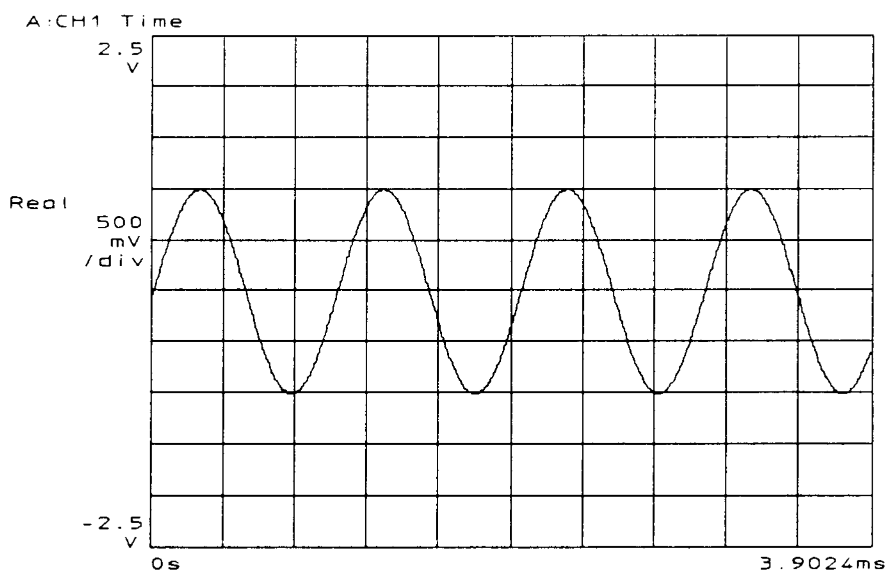
* Available only when analyzer is in two-channel mode

Trigger Setup

After it is armed, the analyzer will start a measurement when the specified trigger conditions are met. You can control where in the cycle triggering occurs by setting up three trigger parameters: level, slope, and delay.

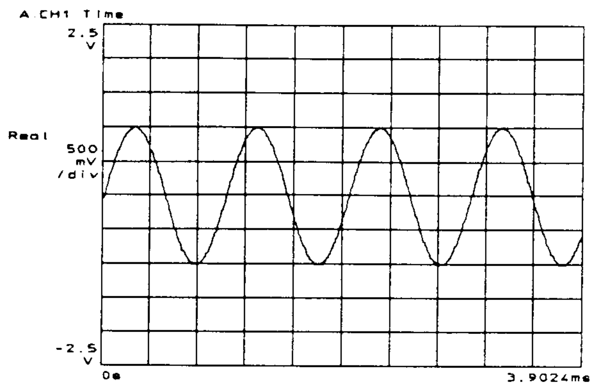
Setting up triggering is often an iterative process. You may have to make several sample measurements, adjusting the trigger parameters as necessary. If the input signal is noisy, false triggers may occur if the trigger level is set too low. You can raise the trigger level to prevent false triggers, but this might cause the analyzer to miss the first part of a transient signal. When you specify a pre-trigger delay, the measurement includes the entire transient signal.

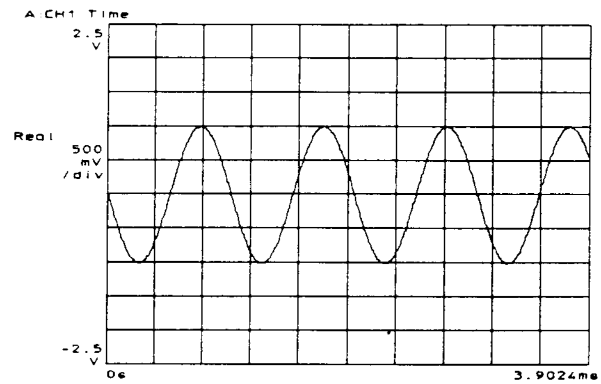
The following examples illustrate the effect of each of the trigger parameters on the measurement. Consider a 1 kHz sine wave with an amplitude of 2 V peak-to-peak. If you specify a level of 0% of full range, slope positive, and no delay, the measurement will start when the signal passes through 0 in a positive direction.



Changing trigger slope

If you change the slope from positive to negative, the display changes as shown in the following traces:

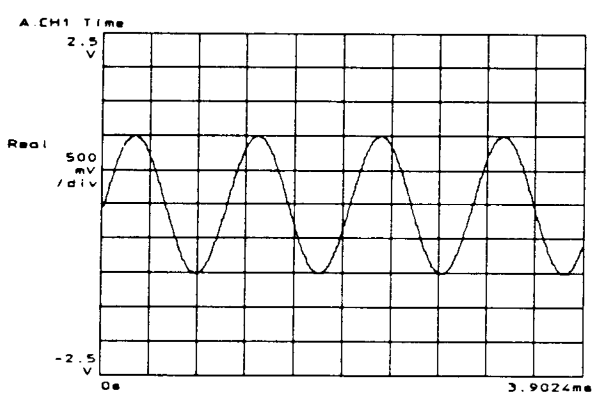


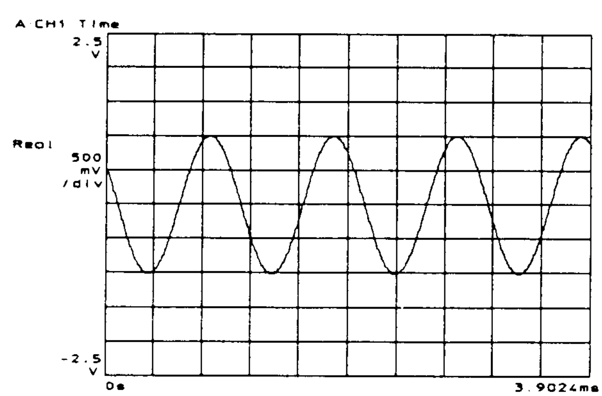


Changing trigger level

If you change the level from 0% to 50% of full range, with the slope positive, the trigger point changes as shown in the following traces:

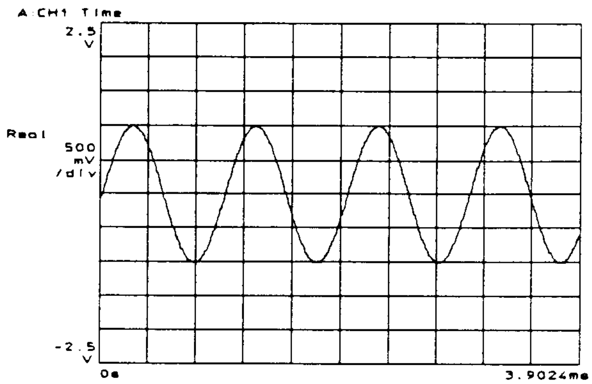
Note Trigger level is set as a percent of the input range. When the range changes, the trigger level also changes.

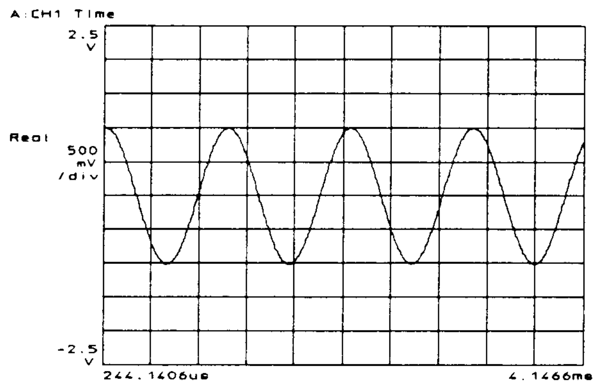




Changing trigger delay

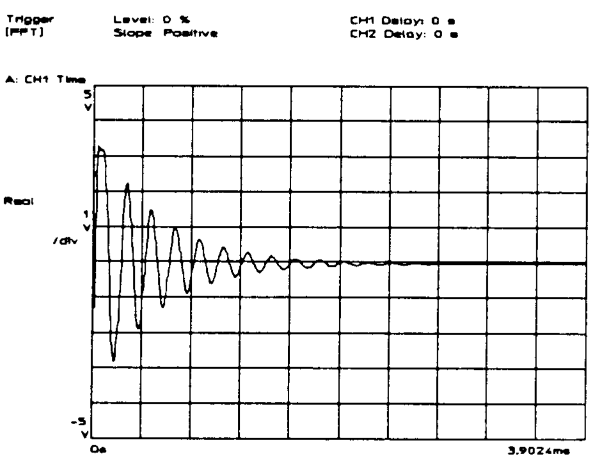
If you specify a level of 0%, slope positive, and change the delay from 0 seconds to 244 μseconds, the measurement will not trigger until 244 μseconds after the trigger conditions are met.

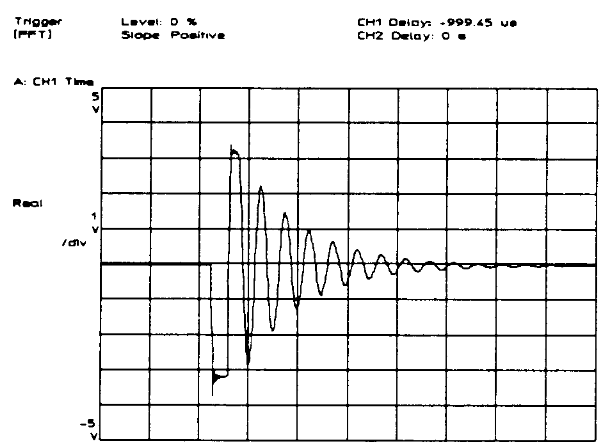




The following illustrations show the effect of pre-trigger delay on a measurement. Pre-trigger delay allows you to start a measurement a specified time *before* the trigger conditions are met. This is very useful for capturing the beginning portions of transients.

The illustration on the left shows a transient signal with no pre-trigger delay. The illustration on the right shows the same transient signal with a 1 msecond pre-trigger delay. Notice how much of the transient signal was missed with no pre-trigger delay.





For detailed information on trigger slope, level, and delay, see the *HP 35665A Operator's Reference* or the analyzer's online help.

6

Averaging

Introduction

The ability to average a series of measurements is extremely useful for many situations. For some measurements, averaging lets you discriminate between random events (noise, for example) and components that are actually part of the signal you want to characterize. Some types of averaging also let you weight "old" data and "new" data differently, to track subtle changes that may occur over a relatively long measurement time.

Keep in mind that some types of averaging are available only with certain instrument modes. Additionally, there are some types of averaging that behave differently depending on the instrument mode you've selected. There are also features—for example, peak hold, a fast averaging, and repeat—that are not technically "averaging" but are still considered part of the averaging process.

General Behavior

An average is a series of combined measurement results that incorporates anywhere from 1 to 9,999,999 averages. As the analyzer makes a series of averages, the results from the last completed measurement are combined, point-by-point, with the previous measurement. These combined results are retained and displayed as they are updated.

To increase the speed of averaged measurements, you can select *fast averaging*. For fast averaging, the analyzer does not update the displayed measurement after each average but instead, displays measurement results only after a certain number of averages. We'll discuss fast averaging later in this chapter.

When the analyzer completes a series of averages, you can restart the measurement or you can continue the measurement. If you restart the measurement, the average count goes back to 0 and begins again. If you continue the measurement, the analyzer adds the next series of averages to the results of the previous measurement—for example, if you've taken 25 averages, continuing the measurement (by pressing [**Pause/Continue**]) adds another 25 averages to make a total of 50 averages for that particular measurement.

Averaging and Overload Conditions

As we mentioned in chapter 4, the analyzer has an *overload reject* feature. This prevents overloaded time records from contaminating an averaged measurement. You can turn on overload reject by pressing [**Avg**] and then toggling to [OVLD REJ **ON** OFF].

When overload reject is *off*, the analyzer displays the **OVLD** message at the bottom of the screen to indicate that an overload has occurred. The **OVLD** message remains visible even when the overloaded signal is removed to indicate that the averaged measurement is contaminated by overloaded time records. Both autorange and fixed input ranges behave the same way when overloaded during an average.

When overload reject is *on*, the analyzer does not make a measurement while the input channel is overloaded. This means that during an averaged measurement, the analyzer stops the average count—suspending the measurement process—when an overload occurs. The analyzer does not resume the average until one of the following happens:

- The input signal is reduced enough to remove the overload condition (either fixed range or autorange input configuration).
- You press [**Start**] to begin a new averaged measurement (autorange input configuration only). The analyzer autoranges to accommodate the overloading input signal. Pressing [**Pause/Cont**] in this case also starts a new averaged measurement.

Averaging and (and related features) available with each instrument mode

	FFT analysis	Octave Analysis	Order Analysis	Swept Sine	Correlation Analysis	Histogram/ Time
RMS	yes				yes	
RMS Exponential	yes				yes	
Vector	yes				yes	
Vector Exponential	yes				yes	
Peak Hold	yes	yes *				
Linear		yes				
Exponential		yes				
Equal Confidence		yes				
Time			yes			
Time Exponential			yes			
Fast Average	yes			yes †	yes	
Settle Time				yes		
Integrate Time				yes		yes
Overlap capability	yes				yes	
Repeat capability	yes	yes	yes		yes	yes
Preview capability	yes					

* Peak Hold in Octave mode is different than in other modes—see text for explanation
† Fast Averaging in Swept Sine mode is different than in other modes—see text for explanation

RMS Averaging

Overview

To make an rms (root-mean-square) average, the analyzer takes the result of the last completed measurement, adds it (point-by-point) to the previous measurement, and takes the rms sum of the two to compete the average. These combined results are retained and displayed as they are updated. (Actually, this description is a simplification—what really happens is that the spectrum is multiplied point-by-point by its conjugate. These results are added, point-by-point. Finally, everything is divided by the number of averages.)

You will find rms averaging useful for a wide variety of measurements. One of the advantages of an rms average is that you do not need to provide a trigger signal. On other FFT spectrum analyzers, rms averaging is sometimes called *normal averaging*, *linear averaging*, or *stable averaging*. The HP 35665A has a type of averaging called *linear averaging*, but this is available only for octave measurements (and is not the same thing as rms averaging).

Special Considerations

It's important to remember that rms averaging does not eliminate noise—it simply produces an approximation of the actual noise level. Increasing the number of rms averages provides a better statistical approximation of the noise, but will not actually reduce the noise.

With rms averaging, the magnitude of individual frequencies also includes noise. So for very small signals, noise can add significantly to that component's magnitude. If you want to reduce that noise, you'll have to use select vector averaging instead.

Vector Averaging

Overview

With vector averaging, the analyzer averages complex values, point-by-point, in the frequency domain. This lowers noise because the real and imaginary components of the random signals are not in phase and therefore cancel each other—increasingly so with each average. Frequency components that are periodic do not cancel and therefore do not diminish with successive averages.

For mechanical applications, vector averaging is often used during vibration measurements to resolve low-level frequency components from background noise.

Vector averaging produces results similar to *time averaging*, a feature found on many FFT analyzers (time averaging means that the analyzer averages all time records first, then performs a single FFT on an averaged time record). Vector averaging accomplishes the same thing as time averaging, since an rms-averaged spectrum derived from a series of vector-averaged spectra is equivalent to a single rms spectrum of time-averaged time records.

Time averaging is available with the HP 35665A, but only with the Order Analysis instrument mode. To learn more about time averaging for this application, see "Time Averaging" later in this chapter.

Special Considerations

Although measurements made with vector averaging have better signal-to-noise ratios than rms averaging, there are some restrictions:

- The input signal must be periodic. In other words, the frequency components you want to measure must repeat with each time record. If these components are not periodic (not in phase with the start of each new time record), their real and imaginary values will cancel and the analyzer will not resolve these components.

- If you select vector averaging, you'll need to provide a trigger signal—from the analyzer's source or from an external signal. Of course, the analyzer will still make a measurement with continuous triggering (no trigger signal), but the amplitude of periodic signals will diminish with each successive average (since even periodic components have random phase with continuous triggering).

Exponential Averaging (RMS or Vector)

Overview

You can select either rms exponential averaging or vector exponential averaging. Both work in a similar fashion. The only difference is that for vector exponential averaging, you'll need to provide a trigger signal. Unlike "normal" averaging, exponential averaging weights new data more than old data. This is useful for tracking data that changes over time.

When using exponential averaging, the number of averages you select determines the weighting of old versus new data—not the total number of averages calculated. As the number of averages increases, new data is weighted less. When starting an exponential average, the analyzer sets N equal to 1 for the first analysis, then sets N equal to 2 for the second analysis, and so on—until N equals the number of averages you've specified.

Note that for the first N averages, all the data is weighed equally (in other words, until you exceed "N" averages, there is no difference between normal averaging and exponential averaging).

To calculate the exponential average, the analyzer uses this formula:

$$\text{next value} = \frac{1}{N}(\text{new value}) + \frac{N-1}{N}(\text{current value})$$

where: N is the weighting factor (the number of averages you've specified).

How it Works

For example, if you set N to 100, the accumulated average (the displayed data) is weighted by 99/100 and the new data by 1/100. You can imagine that for this case, the most recent measurement added to the accumulated average does not change the accumulated average very much. So if you were to change the input signal, the displayed trace would reflect this change very slowly.

Now consider another example. If you set N equal to 5, the accumulated average is weighted by 4/5 and the new data by 1/5. In this case, the new data acquired with a each new measurement can make a considerable difference to the trace. If you were to change the input signal, the displayed trace would reflect this change much more quickly.

Special Considerations

Exponential averaging is easy to use, but there are some things to keep in mind:

- Once you start a measurement using exponential averaging, the measurement continues indefinitely. To stop the average, you must pause the measurement. This is different than normal rms or normal vector averaging, which stops automatically after the specified number of averages.

- With exponential averaging, it's especially important to set the number of averages carefully—if there are too few averages in the measurement, the averaging will not smooth out variances. But if there are too many averages, the analyzer may not track subtle changes occurring within the data.

- Until you accumulate "N" averages, old and new data are weighted equally. In other words, until you exceed "N" averages, there is no difference between normal averaging and exponential averaging.

Peak Hold

With the peak-hold function, the analyzer mathematically compares each data point to its previous peak value. If the data point is larger than its last peak value, the new value is used. This is not the same thing as peak-holding the displayed trace.

Technically, peak-hold averaging is not really a type of averaging, since the results are not mathematically averaged. But it's still considered a type of averaging because it combines the results of several measurements into one final measurement result.

Peak hold is a convenient way to measure the frequency drift of a particular spectral component. You might want to zero the offset marker at this frequency component when you first turn on peak hold—this makes it easier to find the point where the signal drifted *from*.

By the way, do not confuse peak hold with the *peak track* function. Peak hold provides the maximum values at each display point. Peak track simply moves the main marker to follow the largest signal. To learn about Peak track, see chapter 14, "Marker Concepts."

Linear Averaging

Available only with Octave Analysis mode.

Linear averaging weights old and new data records equally to yield the arithmetic mean. The analyzer performs a linear integration over time of the magnitude squared power in each measurement band.

Note

For octave measurements, averaging is done for a specified *time* rather than a particular number of averages. The measurement stops after the specified average time has elapsed.

Exponential Averaging

Exponential averaging for Octave Analysis mode is different than exponential averaging available with other instrument modes.

Unlike linear averaging, exponential averaging weights new data more than old data. This is useful for tracking data that changes over time.

The *average time* you select determines the weighting of old versus new data, not the time during which the analyzer averages. With exponential averaging, the octave measurement continues until you stop it.

The analyzer uses the *average time* to smooth the data. An average time of 0.125 seconds corresponds to an IEC 651 sound level meter "fast" characteristic; 1.0 seconds corresponds to the "slow" characteristic.

Note

Before an exponential measurement is settled, the instantaneous spectrum is displayed. Exponential averaging proceeds after the settling time has elapsed. This behavior is especially noticeable for long time constants (time constants greater than 1 second).

Equal Confidence

Available only with Octave Analysis mode.

For equal confidence the analyzer varies the averaging time for each band so that the relative confidence in the measurement is equal across all bands.

There is a 68 percent probability that the results will be within +/– the specified confidence level of the true mean value. There is a 96 percent probability that the results will be within twice the specified confidence level of the true mean value.

For example, if you specify a confidence level of 2 dB, there is 68 percent confidence that the results will be within 2 dB of the true mean value, and 96 percent confidence that the results will be within 4 dB of the true mean value.

You can specify the confidence level to be 0.25, 0.5, 1, or 2 dB.

Note

Before an equal confidence measurement is settled, the instantaneous spectrum is displayed. Exponential averaging proceeds after the settling time has elapsed. This behavior is especially noticeable for long time constants (time constants greater than 1 second).

Peak Hold (Octave Mode)

The peak hold feature in Octave Analysis mode is different than the peak hold feature in other modes. See the section called "Peak Hold" to learn about peak hold for these other modes.

When you select the peak hold feature in Octave Analysis mode, the analyzer holds the absolute peak power in each displayed octave band. The overall power band (far right band in the display) displays broadband peak power; it has a bandwidth of dc to the value listed in the table below.

Peak hold limits the maximum center band frequency and the broadband peak frequency as shown below:

	1/1 Octave		1/3 Octave		1/12 Octave	
	center	broad	center	broad	center	broad
1-channel	8.0 kHz	25.6 kHz	16.0 kHz	25.6 kHz	5.657 kHz	6.4 kHz
2-channel	4.0 kHz	12.8 kHz	8.0 kHz	12.8 kHz	2.828 kHz	3.2 kHz

Hold Setup Feature

For Octave measurements, you can detect maximum or minimum values that occured within each octave band. You can do this using the [HOLD SETUP] menu.

When you select maximum or minimum, the analyzer displays the maximum or minimum averaged spectrum value for each band. This applies for linear, exponential, and equal confidence octave average types. *It does not affect peak hold (octave) function.*

Note

The [PEAK HOLD] function and the [HOLD SETUP] menu are not related. Do not confuse them. The [HOLD SETUP] affects only the operation of the linear, exponential, and equal confidence octave averaging types.

From the [HOLD SETUP] menu you can do the following:

- Turn off the average hold feature.*
- Hold the maximum averaged spectrum value.
- Hold the minimum averaged spectrum value (this is useful for estimating background noise).

Time Averaging

Time averaging is available only with Order Analysis mode.

For time averaging, the analyzer averages N time records, where N is the specified number of avarages. The averaged time record is not diplayed until the average is complete—so as you increase the number of averages, the display update rate decreases.

Note

When you select time averaging, the analyzer effectively sets the number of waterfall steps to 1. If you pause a measurement with a waterfall displayed, the analyzer displays only one trace.

Time Exponential Averaging

Time exponential averaging is available only with Order Analysis mode.

Conceptually, time exponential averaging is similar to rms and vector exponential averaging available with the analyzer's FFT Analysis mode. Unlike linear (normal) averaging, exponential averaging weights new data more than old. This is useful for tracking data that changes over time.

When using exponential averaging, the number of averages you select determines the weighting of old versus new data—not the total number of averages calculated. As the number of averages increases, new data is weighted less.

To calculate the time exponential average, the analyzer uses this formula:

$$\text{next value} = \frac{1}{N}\,(\text{new value}) + \frac{N-1}{N}\,(\text{current value})$$

where: N is the weighting factor (the number of averages you've specified).

To learn more about exponential averaging, review "Exponential Averaging" earlier in this chapter.

Fast Averaging

You also have a choice of *fast averaging*. Fast averaging isn't really a type of averaging, but rather a way to limit the number of times the analyzer updates a measurement between averages. For a long series of averages, the time savings can be significant.

Swept Sine and Histogram/Time measurements are not averaged like the other instrument modes. However, you can specify both *settle time* and *integrate time* to affect the speed of a swept sine measurement. To learn more, see "Integrate Time" and "Settle Time" later in this chapter.

Settle Time

Available only with Swept Sine mode.

Settle time is the delay between changing the source frequency and starting the measurement at each point. This allows the transient response of the device under test to die out before data collection begins. You can enter the settling time in seconds or as a number of cycles.

Settling time is also useful when you are measuring a device where the output signal is delayed from the input signal. For example, if you are making a swept sine measurement to characterize the performance of a magnetic tape recorder, you must use a settling time to compensate for the delay caused by the distance between the record head and the playback head.

To learn more about swept sine measurements, see chapter 9.

Integrate Time

Available only with Swept Sine mode.

Integration time is the amount of time that each point is measured during a swept sine measurement. Increasing integration time effectively narrows the bandwidth at each measurement point. The result is greater harmonic rejection and increased signal-to-noise ratios but longer measurement times. Setting the integrate time in seconds results in a constant integration scale. Setting the integration time in cycles results in a proportional integration scale; at higher frequencies the same number of cycles occurs in a shorter time.The analyzer actually integrates over an integer multiple of cycles at the measurement frequency. If you enter an integrate time less than 1 complete cycle, the analyzer takes a complete cycle.

Repeat Capability

If you've turned on averaging, the analyzer stops after completing the specified number of averages. For example, if you've specified 25 averages, the analyzer makes 25 measurements and then stops. If you want to make another series of 25 measurements, you must press [**Start**].

There's an easy way to repeat a series of averaged measurements without pressing [**Start**] each time. All you have to do is toggle to [REPEAT **ON** OFF]. Once you turn on repeat, the analyzer starts a new series of averaged measurements after the previous series is finished. To stop the repeated measurements, you can press [**Stop**] or toggle to [REPEAT ON **OFF**].

You can also turn repeat averages to get a set of averages each time the analyzer's trigger is armed. This applies to manual arm, time step arm, or rpm step arm.

Preview

Preview is a feature that lets you reject (or accept) the results of an individual time record so that it doesn't affect your results during an averaged measurement. This is useful for many measurements, including impact testing where it is often necessary to reject poor hammer hits.

Note Preview is available only with the analyzer's FFT Analysis mode.

There are two ways to use preview. If you use manual preview, the analyzer waits for you to accept or reject a time record. If you use timed preview, the analyzer gives you a certain amount of time (adjustable) during which you can accept or reject a time record—if you don't make a decision during this time, the analyzer accepts the time record.

To learn more about using the softkeys in the [PREVIEW SETUP] menu, see the *HP 35665A Operator's Reference.*

Overlap Processing

As the span you select decreases, the corresponding time record length increases (see "Measurement Speed versus Time Record Length" in Chapter 2, "Measurement Basics"). At some point, the time record length and the amount of time the analyzer needs to process each record are equal. If you continue to increase the record length, the FFT processor sits idle after processing the time record (while waiting for the next record to fill). But *overlap processing* allows you to overlap time records and compute the FFT from both previous *and* current time records.

Overlap processing offers several advantages. First of all, it lets you make a faster measurement (particularly with narrow spans/long time records). Overlap processing also reduces statistical variance caused by windowing. For a detailed discussion of overlap processing and real-time bandwidth, see *Hewlett-Packard Application Note 243* (available from your local HP Sales/Service Office).

Overlap processing is set from the averaging menu. To set the amount of overlap, you specify a percentage. You can enter any value from 0 to 99 percent, in 1 percent increments.

Special Considerations:

Overlap Processing can be useful for some measurements, but you should be aware of the following considerations:

- Overlap is not used if you're making triggered measurements. The analyzer must be in the continuous trigger mode.

- The amount of overlap possible varies with the frequency span. For wide spans (with short time records), little or no overlap is possible—the time record is small compared to the time it takes the analyzer to process the time record. For narrow spans (with long time records), considerable overlap is possible—the time record is long compared to the time it takes the analyzer to process the time record.

- The analyzer always uses the requested overlap, if it can meet the processing rate required to do so. The analyzer displays "real time" on its status line (the area over the Trace A title) when it is measuring continuous time records at the requested overlap percentage with no gaps. If the analyzer can't keep up with the requested overlap, the "real time" message disappears and the data is processed with no overlap. For example, the display rate, marker operations, HP-IB activity, or other activities can cause the analyzer to fall out of real time operation.

Overlap Processing and Real-Time Bandwidth

Overlap processing is easy to understand if you relate it to real-time bandwidth (RTBW). Real-time bandwidth is a specification used to characterize the performance of an FFT analyzer. The real-time bandwidth is the frequency span at which the FFT processing time equals the time record length—this means all input data is included in the average (in other words, there is no gap between the end of one time record and the beginning of the next). However, if you increase the span past the real-time bandwidth, the record length becomes shorter than the FFT processing time. Time records are no longer contiguous, and some input data is missed. Therefore, you can overlap records only when measuring below the real-time bandwidth, because the time record length must be longer than the FFT processing time to achieve any overlap.

The actual real-time bandwidth achieved varies with the amount of processing time the analyzer needs. As with overlap processing, this depends on the current frequency span, the type of average selected, and how busy the analyzer is servicing the HP-IB and marker functions and key presses.

The following table shows the maximum overlap available with the HP 35665A, using the default display resolution (400 lines). For 800-line resolution, the percentages are slightly lower. For 200- and 100-line resolution, the percentages are slightly higher. *Note that the following percentages assume rms averaging on, fast averaging on, and the fast averaging update rate equal to the total number of averages.* Additionally, the maximum overlap percentages vary with the number of averages—higher overlap percentages are available when using fewer averages.

	Maximum Overlap Available (400-line resolution)	
Span	**1-channel mode**	**2-channel mode**
12.8 kHz	45 %	N/A
6.4 kHz	75 %	25 %
3.2 kHz	90 %	65 %
1.6 kHz	99 %	85 %
800 Hz	99 %	95 %
400 Hz *	99 %	99 %

* For spans 400 Hz and below, the maximum available overlap is 99 percent

7

Windowing

Introduction

A *window* is a time-domain weighting function applied to the input signal. A window is a filter used to remove signals that are not periodic (and therefore spurious) within the input time record. This makes the input time record appear to be a periodic signal, usually by forcing the amplitude to zero at both ends of the time record.

Depending on the window, the analyzer attenuates different parts of the input time record to prevent *leakage*—a smearing of energy across the frequency spectrum—caused by transforming signals that are not periodic within the time record. Windowing is a concept basic to understanding FFT spectrum analyzers. To learn more, see *Hewlett-Packard Application Note 243* (available from your local HP Sales/Service Office).

You can use the following window types with the FFT Analysis and Order Analysis modes:

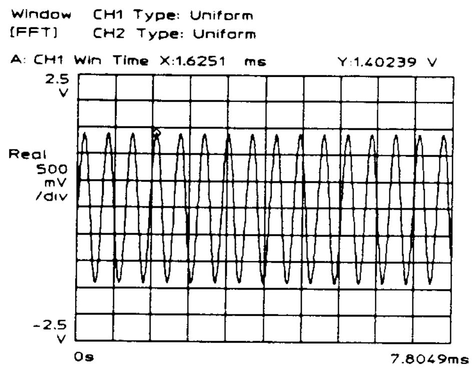
- Hanning.
- Flat Top.
- Uniform.
- Force.
- Exponential.

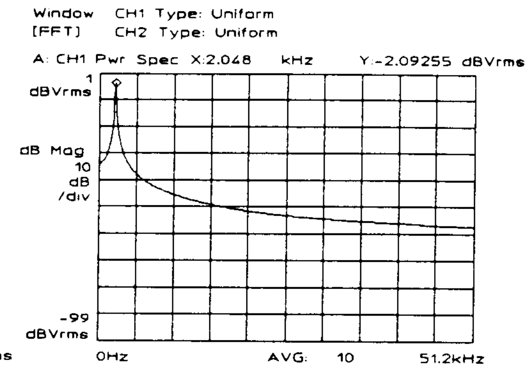
These are standard window types used with most FFT analyzers. Each window offers particular advantages. Because each window type produces different measurement results (just *how* different depends on the characteristics of the input signal and how you trigger on it), you should carefully select a window type appropriate for the measurement you're trying to make. You can see the results of the windowing operation by viewing windowed time data.

Trigger delay is often used to avoid unnecessary attenuation of part of the time record when windowing. For example, suppose you want to examine a transient event and that you are triggering on (or slightly before) the event. Without trigger delay, this transient event occurs at (or near) the beginning of the time record. If you have selected the Flat Top or Hanning windows, the beginning (and end) of each time record is attenuated—thus the transient that you want to examine is attenuated and its shape changed by the effects of the window (to verify this, you should be viewing windowed time data). To center the transient at the center of the time record (where attenuation is the least), you would select the appropriate amount of pre-trigger delay. To learn more about trigger delay, see chapter 5, "Triggering."

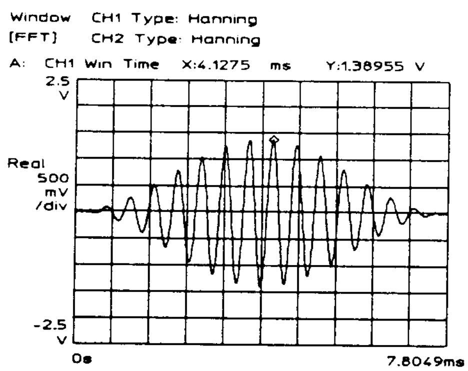
The following figures illustrate the effect of the different windows on a 2 kHz sine wave in the time domain and frequency domain. The instrument mode is FFT analysis. The frequency span is 51.2 kHz.

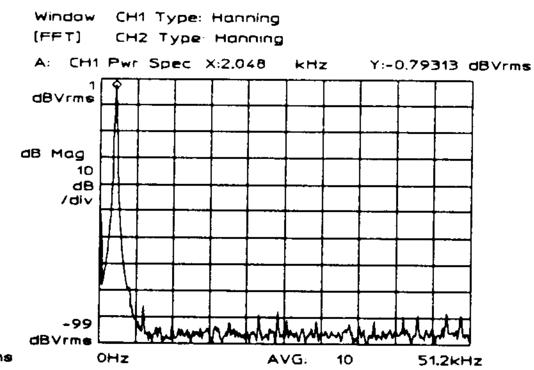
Uniform window:



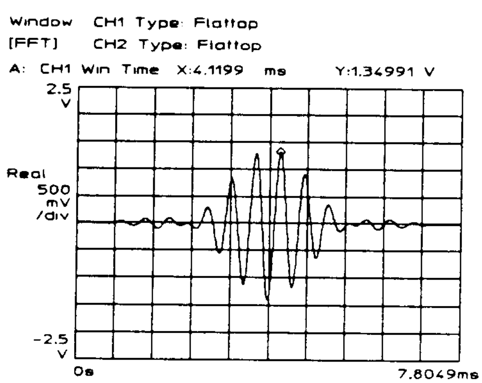


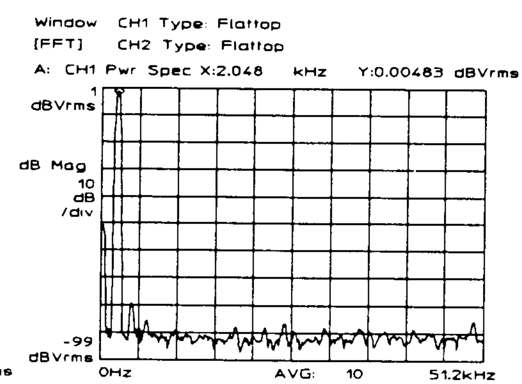
Hanning window:



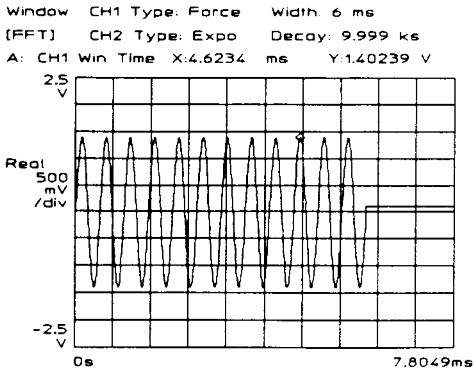


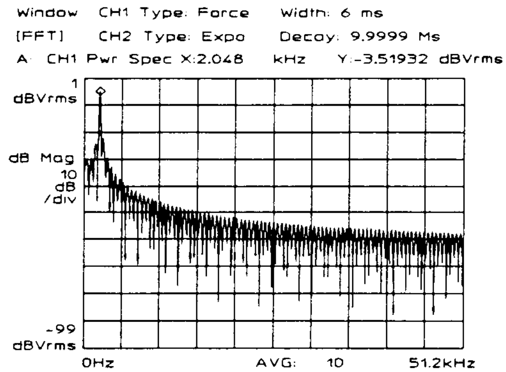
Flat top window:



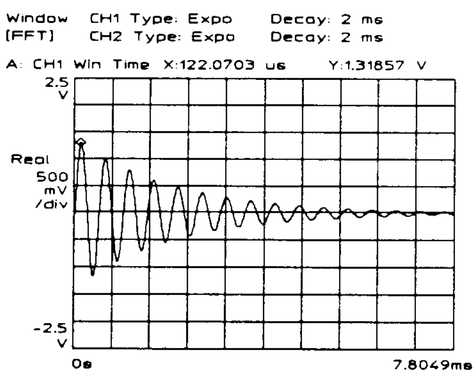


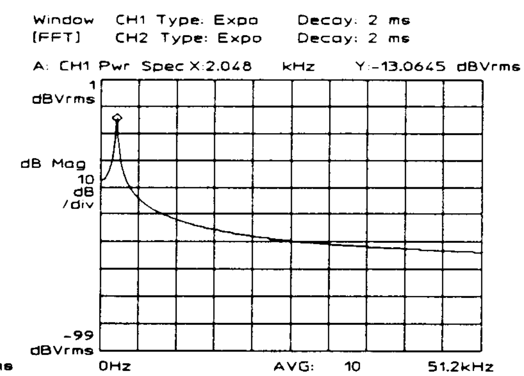
Force window (force width set to 6 ms):



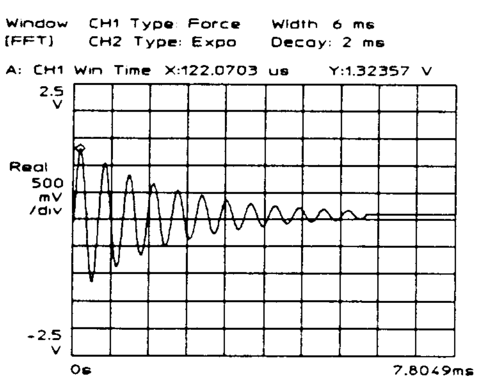


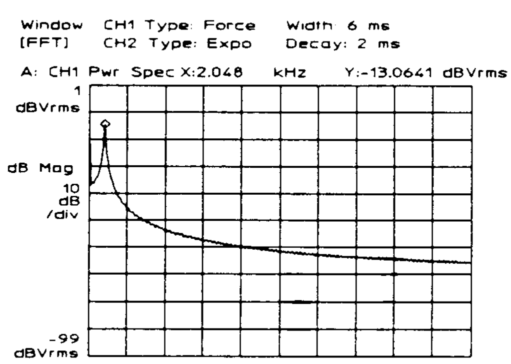
Exponential window (force width set much wider than the time record length, exponential decay set to 2 ms):





Force and exponential windows (force width set to 6 ms, exponential decay set to 2 ms):

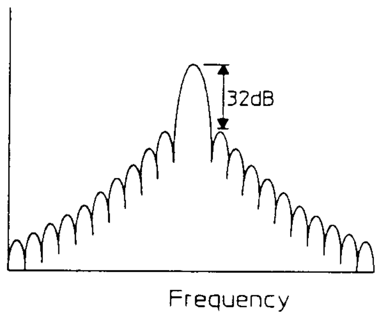


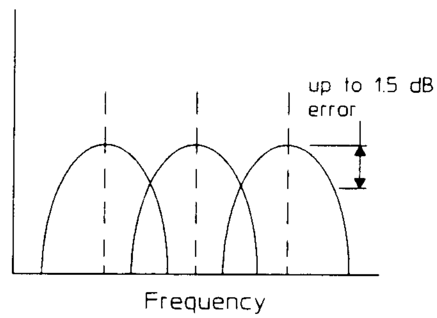


The Hanning Window

The *Hanning* window (sometimes called the *Hann* or *Random* window) attenuates the input signal at both ends of the time record. This forces the signal to appear periodic. The disadvantage of the Hanning window is some amplitude inaccuracy for sinusoidal signals (from 0 to minus 1.5 dB) compared to the Flat Top window. But its advantage is greater frequency resolution.

The Hanning window is the most commonly-used window. It is particularly useful for random noise measurements.

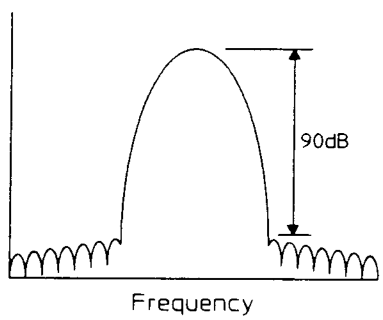


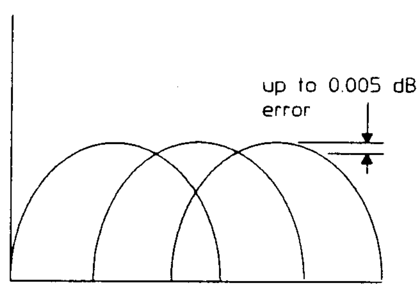


The Flat Top Window.

The *Flat Top* window (sometime called a *sinusoidal* window) compensates for the amplitude inaccuracy of the Hanning window. The flatter shape of the Flat Top window offers greater amplitude accuracy (plus or minus 0.005 dB). But the trade-off is lower frequency resolution.

The Flat Top window is useful when you must measure the amplitude of a particular frequency component with great accuracy—for example, when using a fixed-sine source.



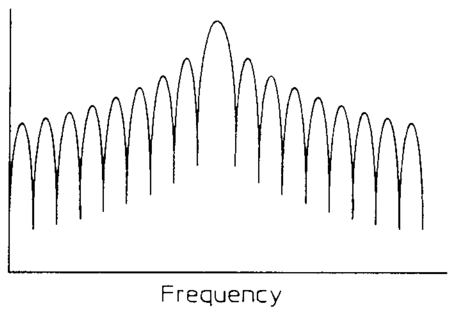


The Uniform Window

The *Uniform window* (sometimes called a *transient* window) has a rectangular shape that weights all parts of the time record equally. In other words, the Uniform window isn't really a window at all.

Because the Uniform window does not force the signal to appear periodic in the time record, it is normally used only with functions that are self-windowing, such a transients and bursts. The Uniform window has an amplitude accuracy uncertainty from 0 to minus 4.0 dB.

For best results with the Uniform window, you should use signal sources that are periodic—for example, a periodic chirp waveform.



The Uniform Window and Periodic Signals: A Brief Demonstration

To better understand why periodic signals should be used with the Uniform window, try this with you HP 35665A:

1) Press [**Preset**].

2) Connect the source to channel 1 and set the source to a 1 volt (rms) sine wave.

3) Compare the effects of the Uniform, Hanning, and Flat Top windows on the 10.24 kHz default sine wave frequency.

4) Now change the sine frequency to 10 kHz. Notice how the spectrum of the 10 kHz signal is smeared with the Uniform window. This is because the 10 kHz signal is not periodic within the time record.

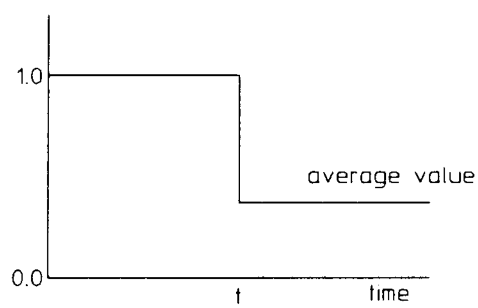
Force Window

The *Force* window passes the first part of the time record and sets the last part to a fixed value. You can specify the width of the window, thus controlling where the fixed level begins. The width you specify determines how much of the signal is passed. Note that the width must be narrower than the time record for the force window to have any effect.

Here are some things to keep in mind when using the force window:

- For the HP 35665A, the analyzer then calculates the average value of the time record's remaining data and sets the time record to this average level.
- The Force window is helpful in impact testing because is removes residual oscillations in lightly damped systems. It is often used with the *Exponential* window (see "Exponential Window").
- Unlike the other windows, many analyzers—such as the HP 35665A—let you apply the Force or Exponential window to each channel individually. This lets you mix the windows in measurements using both input channels, such as frequency response. This application is most commonly used when measuring properties of mechanical structures during impact testing.
- If you apply the Force window to channel 1 and the Exponential window to channel 2, the data for channel 1 is multiplied by both the Force and the Exponential windows.

Hint: *If you use a trigger delay and you want to set the force width using the marker, remember that the time record starts in negative time for pre-triggering. You may have to adjust the window width to allow for this.*



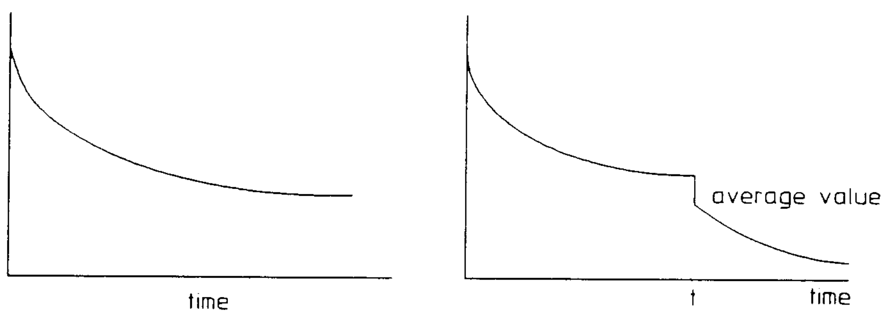
Exponential Window

The *Exponential* window attenuates the input signal at a decaying exponential rate determined by a specified time constant. You can enter a value between 0.1 μS and 9.99 x 10^6 Seconds.

The Exponential window is often used in lightly damped systems with frequency responses that do not decay within one time record.

Here are some things to keep in mind when using the exponential window:

- Generally, the time constant should be set to one-fourth of the time record for the window to be effective.
- This window attenuates the input signal at a decaying exponential rate determined by the specified time constant.
- If you apply the Force window to channel 1 and the Exponential window to channel 2, the data for channel 1 is multiplied by both the Force and the Exponential windows.

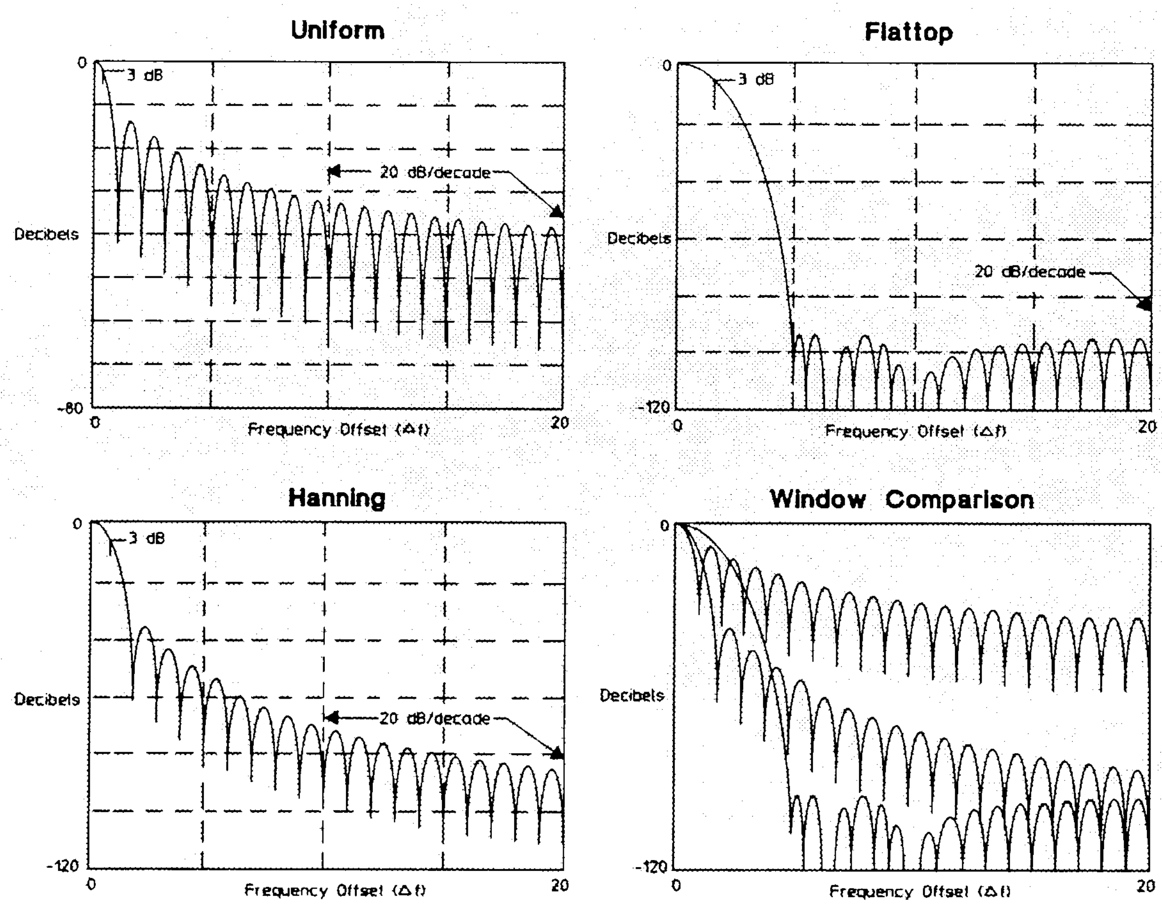


Why is windowing necessary?

The HP 35665A functions as if the input signal were applied to a parallel bank of 401 narrow-band filters. The illustrations here show the frequency-domain response of a single filter when using Uniform, Hanning, or Flat Top windows.

The left side of each illustration represents the center of each filter. Since the filters are symmetrical, we've shown only one side of each filter response (the other side is a mirror image). The horizontal axis indicates the number of frequency bins (display points) offset from the one where the filter is centered.

Think of each drawing as a template. If you position a sine frequency at the exact center of the filter, more of the sine wave's energy appears in the center bin. Some of its energy also appears in other bins. The amount of energy that spills into adjacent bins depends on the type of window you use. Notice how the Hanning window provides better frequency resolution than the Flat Top window. This is because with the Hanning window, less energy spills into nearby bins.



8

FFT Analysis

Introduction

As we discussed in chapter 2, "Measurement Basics," FFT analysis is a fast, easy way to characterize signals in the frequency domain. In FFT analysis mode, the HP 35665A uses digital signal processing to sample the input signal and convert it to the frequency domain. This conversion is done using the *Fast Fourier Transform* (FFT). The FFT is an implementation of the Discrete Fourier Transform, the math algorithm used for transforming data from the time domain to the frequency domain.

The Fourier transform integral converts data from the time domain into the frequency domain. However, this integral assumes the possibility of deriving a mathematical description of the waveform to be transformed—but real-world signals are complex and defy description by a simple equation. The Fast Fourier Transform (FFT) algorithm operates on sampled data, and provides time-to-frequency domain transformations without the need to derive the waveform equation.

Measurement data available with FFT Analysis mode

Power Spec CH1	yes
Power Spec CH2	yes*
Linear Spec CH1	yes
Linear Spec CH2	yes*
Time Channel 1	yes
Time Channel 2	yes*
Windowed Time Channel 1	yes
Windowed Time Channel 2	yes*
Capture Channel 1	yes
Capture Channel 2	yes*
Frequency Response	yes*
Coherence	yes*†
Cross Spectrum	yes*
Orbit	yes*
Math Function	yes
Data Register	yes
Waterfall Register	yes

* Available only when analyzer is in two-channel mode
† Available only when averaging is turned on

Composite Power CH1	
Composite Power CH2	
Order Track CH1	
Order Track CH2	
RPM Profile	
Normalized Variance CH1	
Normalized Variance CH2	
Auto Correlation CH1	
Auto Correlation CH2	
Cross Correlation	
Histogram CH1	
Histogram CH2	
Probability Density Function CH1	
Probability Density Function CH2	
Cumulative Density Function CH1	
Cumulative Density Function CH2	

Measurement Data and the FFT Analysis Mode

With this instrument mode, you have the following measurement data available:

- *Linear Spectrum* (channel 1 or channel 2). This is the data that resulted when the analyzer filtered the most recent input data and performed an FFT on the time record of this sample. Linear spectrum data is the most basic spectrum measurement data available.

- *Power Spectrum* (channel 1 or channel 2). This data is derived from the linear spectrum data. To yield a power spectrum from the input data, the analyzer multiplies the linear spectum data by its complex conjugate and then calculates rms values for each point along the frequency spectrum. Power spectrum data thus provides good rms approximation for noise and other signals. Keep in mind that with power spectrum data, no phase information is available.

- *Time* (channel 1 or channel 2). This is the most recent sampled time record on which the analyzer performed an FFT operation. The time record (and the FFT of this time record) is the basic building block for all subsequent measurement data (the time data appears in the time domain and thus looks like an oscilloscope trace). You can use time data to look at an input signal in the time domain—and use the analyzer's *time parameter markers* to characterize transients, pulses, and other waveforms. When viewing time data, you must set the analyzer to full span to avoid visual distortion of the time trace, undesired filtering, and anti-aliasing artifacts (see related sidebar).

- *Windowed Time* (channel 1 or channel 2). This is similar to time data, but shows the time record after the analyzer has windowed the time time record prior to the FFT process (to learn more about windowing, see chapter 7). Like the time trace, the windowed time trace represents the most recent sampled time record on which the analyzer performed an FFT operation. Windowed time data lets you see the effects of windowing on a particular input signal—this is particularly useful if you are using force or exponential windows, or trigger delays.

- *Capture* (channel 1 or channel 2). This is a time-domain trace that shows the contents of the analyzer's capture buffer. Using the capture buffer is a way to record input data and play it back for later analysis by making measurements from this captured data (think of the capture buffer as a small data acquisition system or instrumentation recorder). To learn more about capture, see chapter 19, "Capture Concepts."

- *Frequency Response.* Essentially, the ratio between the analyzer's source (applied to the device-under-test and also at channel 1) and the output of the device-under-test (measured at channel 2). The HP 35665A measures frequency response using the ratio of the cross spectrum to the input spectrum (channel 1)—a technique called a *tri-spectral average*.

- *Coherence.* Although coherence data appears in the frequency domain like other spectrum measurements, it is somewhat unusual since it has no vertical units. Instead, coherence is measured on a simple linear scale from 0.0 (no coherence) to 1.0 (perfect coherence). The analyzer calculates coherence from a series of averaged frequency response measurements. Coherence shows how much of a network's output was caused by its input—in other words, an "integrity check." Perfect coherence means that all of the power at a particular frequency was caused by the input to the network. No coherence means that the output at a given frequency was caused by something other than the input. Most coherence values are somewhere between these extremes.

- *Cross Spectrum.* This is a measurement of mutual power between channel 1 and channel 2. When viewed with phase coordinates, cross spectrum reveals the relative phase between the two input signals. When viewed with magnitude coordinates, cross spectrum reveals the product of the magnitudes of the two signals. If both signals have a large magnitude, the cross product is large—if both are small, the cross product is small. This makes cross spectrum a sensitive tool for isolating components common to both signals. Cross spectrum is also used to calculate acoustic intensity (acoustic intensity is proportional to the imaginary part of the cross spectrum between two closely-spaced microphones).

- *Orbit.* This is time-domain data that shows the amplitude of channel one versus the amplitude of channel 2. This produces results that are similar to *Lissajous patterns* on an oscilloscope. Orbit data is often used to analyze rotating machinery—typically, to measure shaft runout with two transducers, one at 90 degrees to the other. Orbit data is useful for balancing and understanding rotor dynamics.

- *Math Function.* Math functions let you perform a variety of operations on current (or stored) traces. A math function can contain current input data, a stored trace, or a constant. Math functions are specified with operands and operators in infix (standard algebraic) notation.

- *Data Register* (eight available). Data registers are provided exclusively for intermediate storage of trace data. For example, you can use a data register as temporary storage for a trace during an HP Instrument BASIC program. Or you can use the trace data as part of a math function by specifying this data register (D1 through D8) as an operand. Data registers are also used when you want to view a trace stored on disk—to do this, you simply recall the stored trace file from the appropriate disk and then load it into a data register; then you select this data register for display using the [**Meas Data**] menu. Keep in mind that *all information stored in data registers is volatile and will be lost when you turn off the analyzer*—make sure you've copied all important traces to disk (except the volatile RAM disk) beforehand.

- *Waterfall Register* (eight available). Waterfall registers are provided exclusively for intermediate storage of waterfall-formatted trace data. Waterfall registers are similar to data registers, with an important exception—you cannot use a waterfall register in a math function. Like data registers, you can use a watefall register to view a trace stored on disk—to do this, you simply recall the waterfall trace file from the appropriate disk and then load it into a waterfall register; then you select this register for display using the [**Meas Data**] menu. Keep in mind that *all information stored in waterfall registers is volatile and will be lost when you turn off the analyzer*—make sure you've copied all important waterfall traces to disk (except the volatile RAM disk) beforehand.

Special Considerations for Time Data

Time data is available in FFT Analysis, Order Analysis, Correlation Analysis, and Swept Sine modes. Windowed time data (showing the attenuation of the parts of a time record with window operations) is available for FFT Analysis and Correlation Analysis mode. For Histogram/Time mode, unfiltered time data is available (see chapter 13).

To interpret time data properly, you should know the following:

- Time traces are not calibrated, so they display only an approximate amplitude value. Still, they are very useful since they show input data before the analyzer filters the input data prior to FFT processing.
- Time traces have more resolution than frequency-domain traces. Although the analyzer has a maximum display resolution of 801 points (800 lines) and cannot display data with greater resolution, time data has a 1024-point resolution (unlike a maximum of 801 points for frequency-domain data). You can take advantage of this better resolution by using the axes scale markers (under the [**Scale**] hardkey) to expand a displayed time trace.
- If you set the analyzer to measure full span, the time data you see is the actual input time record. This is raw, unfiltered input data—the signal from which all subsequent measurements are based. You can use this input time record to verify that there is indeed a signal. You can also use the time record to help you manually set the analyzer's input range. For some signals, you may want to turn off the analyzer's anti-alias filters (under the [**Input**] menu) to prevent ringing or distortion of square waves or transients when viewed in the time domain. Make sure, however, that you turn on the anti-alias filter before making measurements in the frequency domain (the anti-alias filters are necessary to ensure a good transformation from the time domain to the frequency domain).
- If you set the instrument to measure a specific bandwidth (something less than full span), the time record shows the raw input data after filtering. This lets you see if there's energy within the selected span. For zoomed time record displays (start frequency not equal to zero), the displayed amplitude is approximately one-half the actual amplitude.
- If the analyzer is making averaged measurements, the most recent time record added to the average is the one displayed. The analyzer does not show a time waveform that is a cumulative average, since all averaging is done after the time data has been transformed to the frequency domain.
- Although the time data looks like an oscilloscope display, the analyzer is not a digital oscilloscope. The time record represents samples of a waveform. The samples have enough information to accurately reconstruct the input signal—but the human eye may not properly perform the reconstruction. In fact, for frequencies that are higher than about ten percent of the frequency span, there will be noticeable visible distortion.

9

Swept Sine Measurements

Swept Sine measurements (Option 1D2) are available only with those HP 35665A analyzers that are equipped with this option.

Introduction

Swept sine measurements are used to make frequency response measurements on a wide variety of test devices. In this mode, the analyzer's source provides a sine wave that "sweeps" through a specified range of frequencies—actually, this sweep is a series of very small discrete steps. You can vary the speed of the sweep, its resolution (how many steps are used to form one sweep), and the direction of the sweep. You can also specify that a sweep have linear-spaced steps or logarithmic-spaced steps.

Although measurement data available in swept sine mode is similar to measurement data in available in FFT mode, there are important differences. Swept sine data is updated point-by-point as the analyzer measures the spectral energy at discrete frequencies—in contrast, measurements made in FFT mode are updated once per measurement.

Why Use Swept Sine Mode?

Though much slower than frequency response measurements made with the analyzer's FFT mode, swept sine measurements nonetheless have several advantages. These advantages include:

- *Increased dynamic range.* Swept sine mode can produce frequency response measurements with a greater dynamic range than comparable FFT measurements. During swept sine measurements, the analyzer can adjust its input range (by autoranging both up and down) to provide an optimal signal-to-noise ratio at each point during the frequency sweep—a technique not possible when measuring with the analyzer's other instrument modes.

- *Better characterization of non-linear devices.* If you make multiple swept sine measurements, each with a different fixed sine level, you can see how distortion changes with different input levels.

- *More efficient excitation of mechanical structures.* In swept sine measurements, all the energy in a frequency response test is concentrated at only one frequency at any given time, unlike measurements made with random noise excitation. This provides for more efficient excitation of the mechanical test structure.

- *Logarithmic sweeping to better characterize audio-frequency devices.* Logarithmic sweeping lets you characterize audio-frequency devices much faster—and with better resolution—than with linear sweeping.

- *Faster measurements than traditional swept sine analyzers.* With the analyzer's *autoresolution* feature, you can make faster swept sine measurements yet still maintain good resolution. When autoresolution is on, the analyzer monitors the slope of the frequency response and changes the resolution accordingly—this means the analyzer sweeps faster in places where the frequency response is relatively flat but slows the sweep when more resolution is needed.

Additional Considerations for Swept Sine Measurements

Keep in mind that a device-under-test exhibits both a transient response and a steady-state response. During a swept sine measurement, the analyzer's sine source jumps from one step to another to form a complete sweep. At each frequency change, the device-under-test first produces a transient response. As the transient settles out, the steady-state response dominates. To exclude the transient response, the analyzer waits momentarily until the test device settles (this delay is adjustable). This is one of the reasons why a swept sine measurement takes much longer than a frequency response measurement made in the analyzer's FFT mode.

Ramp Rate

You can also specify a *ramp rate* to determine how fast the swept source changes when you start, stop, pause, and continue a swept sine measurement. The source also ramps to a different level if you change the source level during a swept sine measurement or if the autolevel algorithm changes the level.

Autolevel

In Swept Sine mode, you can use an *autolevel* feature that lets the analyzer adjust the source output level to keep the amplitude of one input channel within a specified range. When autolevel is off, the source has a constant amplitude (level) at all measurement points.

When autolevel is on, the amplitude at the measurement frequency is monitored. At each measurement point, the source amplitude is adjusted until the non-reference input amplitude is within a specified tolerance band around the reference level. You can specify the reference channel, reference level, reference tolerance, the maximum source output, and the maximum input level in the autolevel setup menu.

Swept Sine Mode: Another Way to Make Frequency Response Measurements

Introduction

You can make frequency response measurements with either the analyzer's FFT mode or with the Swept Sine mode. Both measurement techniques have their advantages. In this brief introduction, we've shown how swept sine mode may be preferred for certain measurement situations.

The primary objective of a swept sine test is to characterize the gain and phase of a test device by measuring only the fundamental component of the stimulus signal and only the fundamental component of the response signal. Swept sine data is updated point-by-point as the analyzer measures the spectral energy at discrete frequencies. This is the result of the individual measurements that the analyzer makes to evaluate, one-by-one, the discrete frequencies that form a complete sweep.

Out-of-band signals—such as distortion component generated by a non-linear test device—are not measured. This is because swept measurements do not reveal spectral components that were detected at frequencies other than the one at which the analyzer's was "tuned" during the sweep.

In contrast, measurements made in FFT mode are updated once per measurement. When you view a frequency response measurement made the analyzer's FFT mode, you are viewing all spectral components that were present when the analyzer sampled the input signal. Typically, the device-under-test is stimulated by the analyzer's noise or chirp (a very fast sine sweep) source. This provides the necessary excitation across the frequency range of interest during the relatively short time that the analyzer must sample the output of the device-under-test. (To learn more about the FFT process, see chapter 3, "Measurement Basics.")

Time-Domain Integration

Traditionally, swept sine measurements were done with instruments that used analog tracking filters to measure a narrow frequency band centered around the frequency of the swept sine source. As the sine source changed, so did the center frequency of the tracking filter.

The HP 35665A makes swept sine measurements by sweeping its sine source and using time-domain integration to emulate a tracking filter. As each integration cycle concludes, the analyzer adds another point to the frequency response display. This process continues during the course of a sweep, converting time-domain input data to the frequency domain.

To achieve the narrow filter bandwidths required for swept sine measurements, the HP 35665A uses the Fast Fourier Transform to evaluate the energy within a narrow frequency span. The transform is evaluated at multiple points during a sweep, with the center frequency of the FFT analysis corresponding to the frequency of the swept sine source. This technique emulates a tracking bandpass filter with very narrow bandwidths, very good harmonic rejection, and excellent dc rejection.

Measurement data available with Swept Sine mode

Power Spec CH1	
Power Spec CH2	
Linear Spec CH1	yes
Linear Spec CH2	yes
Time Channel 1	yes
Time Channel 2	yes
Windowed Time Channel 1	
Windowed Time Channel 2	
Capture Channel 1	
Capture Channel 2	
Frequency Response	yes
Coherence	
Cross Spectrum	Yes
Orbit	
Math Function	yes
Data Register	yes
Waterfall Register	yes

Composite Power CH1	
Composite Power CH2	
Order Track CH1	
Order Track CH2	
RPM Profile	
Normalized Variance CH1	yes
Normalized Variance CH2	yes
Auto Correlation CH1	
Auto Correlation CH2	
Cross Correlation	
Histogram CH1	
Histogram CH2	
Probability Density Function CH1	
Probability Density Function CH2	
Cumulative Density Function CH1	
Cumulative Density Function \CH2	

Measurement Data and the Swept Sine Mode

Although measurement data available in Swept Sine mode is similar to measurement data in available in FFT mode, there are important differences. Swept sine data is updated point-by-point as the analyzer measures the spectral energy at discrete frequencies—in contrast, measurements made in FFT mode are updated once per measurement.

With swept sine mode, you have the following measurement data available:

- *Linear Spectrum* (channel 1 or channel 2). This shows the swept linear spectrum for either channel 1 or channel 2. As the analyzer makes a measurement, the linear spectrum trace is updated point-by-point as the analyzer sweeps through each discrete frequency.

- *Time* (channel 1 or channel 2). This is the most recent time-domain representation of either channel 1 or channel 2. The time trace is updated as the analyzer sweeps through each discrete frequency.

- *Frequency Response.* The analyzer makes this type of measurement by dividing the channel 2 linear spectrum by the channel 1 linear spectrum. The frequency response trace is updated point-by-point as the analyzer steps through each discrete frequencies in a single swept measurement. *Keep in mind that swept frequency response measurements do not reveal spectral components that were detected at frequencies other than the one at which the analyzer's was "tuned" during the sweep.*

- *Cross Spectrum.* This is a measurement of mutual power between channel 1 and channel 2. When viewed with phase coordinates, cross spectrum reveals the relative phase between the two input signals. When viewed with magnitude coordinates, cross spectrum reveals the product of the magnitudes of the two signals. If both signals have a large magnitude, the cross product is large—if both are small, the cross product is small. This makes cross spectrum a sensitive tool for isolating major components common to both signals.

- *Normalized Variance* (channel 1 or channel 2). This is an indicator of the noise power remaining in the signal after a certain number of integration cycles have completed. Variance values equal to 1 indicate that the noise component was successfully averaged out of the signal. Variance values less than 1 indicate the level of noise power remaining in the signal after the integration process (to improve variance, you can increase the number of integration cycles, thereby improving the signal-to-noise ratio). To learn more about integration (and integration cycles), see the sidebar on "Swept Sine Mode: Another Way to Make Frequency Response Measurements" earlier in this chapter.

- *Math Function.* Math functions let you perform a variety of operations on current (or stored) traces. A math function can contain current input data, a stored trace, or a constant. Math functions are specified with operands and operators in infix (standard algebraic) notation.

- *Data Register* (eight available). Data registers are provided exclusively for intermediate storage of trace data. For example, you can use a data register as temporary storage for a trace during an HP Instrument BASIC program. Or you can use the trace data as part of a math function by specifying this data register (D1 through D8) as an operand. Data registers are also used when you want to view a trace stored on disk—to do this, you simply recall the stored trace file from the appropriate disk and then load it into a data register; then you select this data register for display using the [**Meas Data**] menu. Keep in mind that *all information stored in data registers is volatile and will be lost when you turn off the analyzer*—make sure you've copied all important traces to disk (except the volatile RAM disk) beforehand.

- *Waterfall Register* (eight available). Waterfall registers are provided exclusively for intermediate storage of waterfall-formatted trace data. Waterfall registers are similar to data registers, with an important exception—you cannot use a waterfall register in a math function. Like data registers, you can use a watefall register to view a trace stored on disk—to do this, you simply recall the waterfall trace file from the appropriate disk and then load it into a waterfall register; then you select this register for display using the [**Meas Data**] menu. Keep in mind that *all information stored in waterfall registers is volatile and will be lost when you turn off the analyzer*—make sure you've copied all important waterfall traces to disk (except the volatile RAM disk) beforehand.

10

Octave Analysis

Octave Analysis (Option 1D1) is available only with those HP 35665A analyzers that are equipped with this option.

Introduction

With increasing noise regulations, sound level tests have become mandatory in the development of many products, including automobiles, airplanes, and office equipment. Regulations often require testing for sound pressure, spatial characterization of radiated noise, or noise source identification. The analyzer's full-octave and third-octave filters conform to the specifications of ANSI IEC 225-1966, DIN 45651, and ANSI S1.11-1986.

In Octave Analysis mode, the HP 35665A Dynamic Signal Analyzer makes real-time third octave (1/3) measurements—measurements that are required for many types of acoustic measurements. Additionally, the analyzer can make real-time full octave (1/1) measurements and real-time twelfth octave (1/12) measurements.

A-, B-, and C-weighting functions are used with octave measurements to better approximate the frequency response of the human ear. With the HP 35665A, you can perform weighting using either the analyzer's built-in hardware A-weighting filter or using the weighting functions available as math operations.

Customary Units for Acoustic Measurement

The basic unit of sound pressure is an SI-derived unit called the Pascal (Pa). One Pascal is defined as one Newton per meter squared. However, most acoustic measurements are not expressed directly in Pascals. Rather, they are made in units of SPL (Sound Pressure Level) or Sound Level (SPL units that are weighted to approximate the frequency response of the human ear).

SPL is a ratio of the measured pressure to a particular reference pressure, expressed on a logarithmic scale. The reference pressure is usually 20 microPascals—the threshold of human hearing. Because SPL measurements are relative, this reference level is considered to be 0 dB. In contrast, an SPL of 60 dB (0.02 Pascals) is a thousand times greater than the threshold of hearing.

As useful a unit as SPL is, it does not take into account the frequency response of the human ear. The human ear does not detect all audible frequencies equally. Although a microphone used for SPL measurements will have a flat frequency response out to 20 kHz, the human ear is more sensitive to frequencies in the 500 Hz to 10 kHz range. Outside of this range, the human ear has reduced sensitivity.

To simulate how the human ear perceives sound, SPL measurements must be weighted by a function that approximates the frequency response of the human ear. Sound Level measurements are SPL measurements that are made with this type of weighting function. The American National Standards Institute (ANSI) has standardized several weighting functions. The most widely used is A-weighting. This explains why most Sound Level measurements are expressed in units of dBA (the suffix "A" implies the use of an A-weighting filter).

Octave Overview

Before the development of narrowband spectrum analyzers (for example, real-time FFT spectrum analyzers) acoustic analysis required a special kind of instrument—the third octave analyzer. This instrument uses a bank a parallel filters, each with a wider bandwidth than the preceding filter. The filters are spaced this way for the same reason that piano keys are spaced 12 to an octave—because the human ear perceives frequency proportionally. A change of one octave represents a doubling (or halving) of a particular frequency. This is a logarithmic relationship, not a linear one. For this reason, octave measurements are always displayed with a logarithmic X-axis.

Although measurements made with narrowband analyzers provide better frequency resolution than third octave analyzers, third octave analysis (with A-weighting) is the traditional tool for acoustic analysis. In fact, many noise regulations are still specified in terms of third octave measurements. In addition to the traditional third octave mode, the HP 35665A also makes full octave (1/1) measurements and twelfth octave (1/12) measurements.

You can also use the analyzer's FFT mode to make acoustic measurements (A-weighting is available with FFT mode as well). This can be useful for measurement situations that require greater frequency resolution than that available with the analyzer's Octave Analysis mode. For example, while third octave measurements show how the human ear perceives the frequency content of noise, they do not have enough resolution to reveal the exact spectral components of the noise. To diagnose the cause of a noise problem, you may need to use the analyzer's FFT mode.

A-Weighting With the Hardware Filter

Each input channel has its own hardware A-weight filter. To turn on the filter, press [**Input**] and press [CHANNEL 1 SETUP] or [CHANNEL 2 SETUP]. Then toggle to [A WT FLTR **ON** OFF].

A-, B-, and C-Weighting With Math Operations

Weighting functions as implemented in math operations are defined as:

$$\text{A-weight} = \frac{K_1 \times K_3 \times (j\omega)^4}{(j\omega + f_1)^2 \times (j\omega + f_4)^2 \times (j\omega + f_2) \times (j\omega + f_3)}$$

$$\text{B-weight} = \frac{K_1 \times K_2 \times (j\omega)^3}{(j\omega + f_1)^2 \times (j\omega + f_4)^2 \times (j\omega + f_5)}$$

$$\text{C-weight} = \frac{K_1 \times (j\omega)^2}{(j\omega + f_1)^2 \times (j\omega + f_4)^2}$$

where:

$K_1 = 149762512.0$
$K_2 = 1.012481605$
$K_3 = 1.249935598$

$f_1 = 20.598997$ (Hz)
$f_2 = 107.65265$ (Hz)
$f_3 = 737.86225$ (Hz)
$f_4 = 12194.220$ (Hz)
$f_5 = 158.48932$ (Hz)

General Behavior for Math Weighting Functions

The weighting functions shown here are the ones that the analyzer applies to the operand if you select A-, B-, or C-weighting as a math operation. Keep in mind that these functions have phase characteristics.

- When the operand contains both real and power data (for example, power spectrum), this math function multiplies the magnitude square of weighting function to the data.

- When the operand contains both real and linear data, the magnitude of the weighting function is multiplied.

- When the operand contains complex data, the complex weighting function is applied to the data in order to simulate the A-weighting filter. This causes a phase shift. Be careful not to do math like AWEIGHT(CSPEC), since this changes the original phase characteristics of your data. A possible workaround to this problem is to generate a magnitude only A-weighting curve and multiply it to the cross spectrum. For example, setting F1 = AWEIGHT(K1), where K1 is a unity constant, generates a magnitude A-weighting function. If you want to apply this weighting to the cross spectrum, set F1 = AWEIGHT(K1), F2 = F1 * F1 * CSPEC, and K1 = 1.0.

Measurement data available with Octave Analysis mode

Power Spec CH1	yes	Composite Power CH1	
Power Spec CH2	yes*	Composite Power CH2	
Linear Spec CH1		Order Track CH1	
Linear Spec CH2		Order Track CH2	
Time Channel 1		RPM Profile	
Time Channel 2		Normalized Variance CH1	
Windowed Time Channel 1		Normalized Variance CH2	
Windowed Time Channel 2		Auto Correlation CH1	
Capture Channel 1	yes	Auto Correlation CH2	
Capture Channel 2	yes*	Cross Correlation	
Frequency Response		Histogram CH1	
Coherence		Histogram CH2	
Cross Spectrum		Probability Density Function CH1	
Orbit		Probability Density Function CH2	
Math Function	yes	Cumulative Density Function CH1	
Data Register	yes	Cumulative Density Function CH2	
Waterfall Register	yes		

* Available only when analyzer is in two-channel mode

Measurement Data and the Octave Analysis Mode

With this instrument mode, you have the following measurement data available:

- *Power Spectrum* (channel 1 or channel 2). This is a real-time, rms-averaged spectrum of the power detected for each octave band. Depending on the frequency configuration you've selected, the analyzer meets either ANSI Class III (1/3 octave) or ANSI Class II (1/1 octave), as well as DIN standards for filter center frequency and band shape. The HP 35665A also offers a twelfth-octave (1/12) configuration.

- *Capture* (channel 1 or channel 2). This is a time-domain trace that shows the contents of the analyzer's capture buffer. Using the capture buffer is a way to record input data and play it back for later analysis by making measurements from this captured data (think of the capture buffer as a small data acquisition system or instrumentation recorder). To learn more about capture, see chapter 20, "Capture Concepts."

- *Math Function*. Math functions let you perform a variety of operations on current (or stored) traces. A math function can contain current input data, a stored trace, or a constant. Math functions are specified with operands and operators in infix (standard algebraic) notation.

- *Data Register* (eight available). Data registers are provided exclusively for intermediate storage of trace data. For example, you can use a data register as temporary storage for a trace during an HP Instrument BASIC program. Or you can use the trace data as part of a math function by specifying this data register (D1 through D8) as an operand. Data registers are also used when you want to view a trace stored on disk—to do this, you simply recall the stored trace file from the appropriate disk and then load it into a data register; then you select this data register for display using the [**Meas Data**] menu. Keep in mind that *all information stored in data registers is volatile and will be lost when you turn off the analyzer*—make sure you've copied all important traces to disk (except the volatile RAM disk) beforehand.

- *Waterfall Register* (eight available). Waterfall registers are provided exclusively for intermediate storage of waterfall-formatted trace data. Waterfall registers are similar to data registers, with an important exception—you cannot use a waterfall register in a math function. Like data registers, you can use a watefall register to view a trace stored on disk—to do this, you simply recall the waterfall trace file from the appropriate disk and then load it into a waterfall register; then you select this register for display using the [**Meas Data**] menu. Keep in mind that *all information stored in waterfall registers is volatile and will be lost when you turn off the analyzer*—make sure you've copied all important waterfall traces to disk (except the volatile RAM disk) beforehand.

Center Frequency

When you move the marker from octave band to octave band, the marker readout indicates the nominal center frequency for that particular octave filter. For example, in third octave, the center frequencies might be 10 Hz, 12.5 Hz, 16 Hz, 20 Hz, 25 Hz, and so on. However, the exact center frequency (CF) of each filters is determined by the following equations:

Full-Octave Center Frequencies

There are 11 third-order Butterworth filters for full-octave resolution, with start frequencies that can range from 63 mHz to 16 Hz and stop frequencies from 63 Hz to 16 kHz. The default span is 16 Hz to 16 kHz.

$$CF = 1000 \times 2^{n}$$

where: $n = -14, -13, -12, \ldots -1, 0, 1, 2, 3, 4, 5$

Third-Octave Center Frequencies

There are 33 third-order Butterworth filters for third-octave resolution, with start frequencies that can range from 80 mHz to 10 Hz and stop frequencies from 125 Hz to 32 kHz. The default span is 20 Hz to 32 kHz (one-channel operation) or 10 Hz to 16 kHz (two-channel operation).

$$CF = 1000 \times 2^{\left(\frac{n-30}{3}\right)}$$

where: n = the ANSI band number, where $-11 \leq n \leq 45$

Twelfth-Octave Center Frequencies

There are 132 third-order Butterworth filters for twelfth-octave resolution, with start frequencies that can range from 99.7 mHz to 6.3825 Hz and stop frequencies from 192.8 Hz to 12.34 kHz. The default span is 6.3285 Hz to 12.34 kHz (one-channel operation) or 3.19 Hz to 6.17 kHz (two-channel operation).

$$CF = 1000 \times 2^{\left(\frac{1}{24}\right)} \times 2^{\left(\frac{n}{12}\right)}$$

where: n = the ANSI band number, where $-160 \leq n \leq 43$

Octave Settle Time

It's important to relate the timebase of real-time octave measurements to the start of the measurement, particularly with respect to waterfall memory. Here's how to interpret the time offset of the first octave spectrum in waterfall memory relative to the following:

- Pressing [**Start**] to begin a measurement (freerun trigger)
- External trigger
- HP-IB trigger

The settling time of the octave filters are determined by the bandwidth of the lowest frequency filter in the set. To determine the center frequency of a particular octave filter, move the marker to the lowest octave band—the center frequency for that filter is indicated by the marker readout.

$$BW = \left[CF \times 2^{\frac{1}{(2n)}}\right] - \left[\frac{CF}{2^{\frac{1}{(2n)}}}\right]$$

$$ST = \frac{5.0}{BW}$$

where:

n = 1 (full octave)
n = 3 (third octave)
n = 12 (twelfth octave)

CF = center frequency (see "Center Frequency")
ST = settling time
BW = bandwidth

Other Considerations

In addition to settle time, several measurement parameters can interact to affect the *time scan value* of the first trace saved in the waterfall (you can display the time scan value if you press [SUPLMENTL INFO] under the [**Marker Fctn**] menu).

Before we describe the different scenarios under "Determining the First Valid Waterfall Trace Time," keep in mind the following:

- No data is saved in the waterfall until the settle time has elapsed (this is true despite the fact that for exponential and equal confidence averaging, the analyzer displays pre-settled data). If you stop a measurement before it settles, only a single, non-settled trace is saved to the waterfall.

- In Automatic Arm mode, the waterfall data can wrap (overwrite itself) if the number of selected *waterfall steps* is smaller than the number of traces displayed since settling (to learn about waterfall steps, see the *HP 35665A Operator's Reference*). To avoid this, set the waterfall steps to a large number—for example, 1000—and then pause the measurement before the waterfall is filled.

Determining the First Valid Waterfall Trace Time

Only after an octave measurement has settled does the analyzer begin putting traces in the waterfall buffer. For the purpose of this discussion, we'll call the time of this first waterfall trace FT.

The absolute error of FT depends on the trigger mode. For external trigger, the time ambiguity is 0.5 milliseconds. For HP-IB trigger, the time ambiguity is less than 2 milliseconds, provided the system is in a quiescent state before the trigger is received.

The equations use a parameter called "step." The best way to determine the value of "step" is to measure the waterfall update time for the frequency range of interest while using *automatic arm* and *exponential average*. Use the slice markers to put a slice into a data register, then look at the delta time between traces. For the highest frequency range 31.5 kHz, the step value is 15.625 milliseconds. For other stop frequencies, the step time doubles for each lower octave (31.25 milliseconds at 16 kHz stop freq, and so on).

Two operators are used in the equations that may need some explanation—"floor()" returns the largest integer less than or equal to the passed floating point value. The other is "ceil()"—this returns the smallest integer not less than the passed value. And example of these two operators would be floor(5.2)=5, ceil(5.2)=6.

Automatic Arm, Exponential or Equal Confidence Averaging:

$$FT = \left\{\left[\mathrm{floor}\left(\frac{ST}{step}\right) + 3\right] \times step\right\} + 1\ step$$

where: FT = first valid waterfall trace displayed
ST = settle time (see "Octave settle time")

The waterfall trace is tagged with the time of an instantaneaous snapshot of the exponential average at the end of one step.

Example: 20 Hz to 31.5 kHz, 1/3 octave, BW (20) = 4.63 Hz, ST = 1.08 seconds, step = 0.015625 seconds. Therefore, FT = 1.14 seconds.

Automatic Arm, Linear Averaging:

$$FT = \left\{\left[\mathrm{floor}\left(\frac{ST}{step}\right) + 3\right] \times step\right\} + LAT$$

where: FT = first valid waterfall trace displayed
ST = settle time (see "Octave settle time")
LAT = linear average time

The waterfall trace is tagged with the time from start of measurement that occured at the end of the linear average.

Time Step Arm, Exponential or Equal Confidence Averaging:

$$FT = \left\{\left[\mathrm{floor}\left(\frac{ST}{step}\right) + 3\right] \times step\right\} + TS$$

where: FT = first valid waterfall trace displayed
ST = settle time (see "Octave settle time")
TS = time step size

The waterfall trace is tagged with the time of an instantaneaous snapshot of the exponential average at the end of the time step.

Time Step Arm, Linear Averaging:

If TS (time step) ≤ FT, then: $FT = \left\{\left[\mathrm{floor}\left(\frac{ST}{step}\right) + 3\right] \times step\right\} + LAT$

If TS (time step) > FT, then: $FT = \left\{\left[\mathrm{ceil}\left(\frac{ST}{step}\right)\right] \times step\right\} + LAT$

where: FT = first valid waterfall trace displayed
ST = settle time (see "Octave settle time")
LAT = linear average time

The waterfall trace is tagged with the time from start of measurement that occured at the end of the linear average.

RPM Step Arm, Exponential or Equal Confidence Averaging:

The waterfall has a scan axis in rpm, rather than time. The measurement must be settled before the first rpm arm point is obtained. Otherwise, the measurement is armed but no data is saved.

Keep in mind that for exponential and equal confidence averaging, the analyzer takes a "snapshot" of the exponential average at the rpm arm point.

If the "step" time is longer than the time it takes to ramp between two rpm arm points, rpm steps will be missed.

RPM Step Arm, Linear Averaging:

The linear average begins at the rpm arm point, such as the rpm value in the waterfall represents the beginning of the linear average. If the linear average is too long, or the rpm ramp rate too high, rpm steps may be missed.

The measurement must be settled before the first rpm arm point occurs. Otherwise, the measurement is armed but no data is saved.

11

Order Analysis

Order Analysis (Option 1D0) is available only with those HP 35665A analyzers that are equipped with this option.

Introduction

Order analysis (of which *order tracking* is a subset) is a set of specialized measurement techniques often used when making vibration measurements on rotating machinery. With the Order Analysis option, you can make rpm profiles, order maps, and run-up/run-down measurements with the HP 35665A.

To make order measurements, you must properly synchronize a spectrum analyzer to the rotation of the machine under test. This provides measurements results calibrated to the speed of the test device. When order tracking is on, the X-axis is displayed as *rpm*—when order tracking is off, the X-axis is displayed in *orders*.

Though not all rotating machinery measurements are made with order analysis, it is extremely useful when you need to track individual spectral components to the speed of a particular piece of machinery. In the past, making order measurements was cumbersome. For older spectrum analyzers, you needed to use additional test equipment to make these measurements—usually, a ratio synthesizer and a set of tracking filters—and that often introduced undesired time delays, bandwidth restrictions, and phase noise.

However, the HP 35665A Dynamic Signal Analyzer has a built-in order algorithm that lets you make these measurements *without external equipment.*

Why Use Order Analysis?

Unlike other spectrum measurements, vibration analysis of rotating machinery often requires synchronization with the rotating device to make meaningful measurements. In many measurement situations, knowing the exact frequency of a particular spectral component is not nearly so useful as knowing its occurrence relative to the rotation of the test device.

When using order analysis, the HP 35665A does not display a spectrum with the normal X-axis (calibrated in frequency). Rather, the X-axis is calibrated in "orders" or in rpm. When order tracking is *off*, the X-axis readout is in orders (one order represents one revolution of the test device; two orders would be twice that speed, and so on). When order tracking is *on*, the X-axis readout is in rpm.

The advantage of using order analysis is that the exact speed of the test device is not critical. This is extremely important since it lets you compare the results of several vibration measurements made on the same piece of machinery *even if the machine was not running at exactly the same speed each time.* This makes it much easier to track differences between "baseline" and "current" vibration spectra for a particular machine.

Vibration analysis for rotating machinery is a complex topic and one well beyond the scope of this *Concepts Guide*. You can find applications information in numerous Hewlett-Packard application notes. These are available from your local HP Sales and Service Office. In particular, you might want to request some of the following literature:

- AN 243-1: *Effective Machinery Maintenance Using Vibration Analysis.*
- Potter, Ron and Gribler, Mike. "Computed Order Tracking Obsoletes Older Methods." Reprint from *Proceedings of the 1989 Noise and Vibration Conference*, SAE Technical Paper Series (reprint number 891131).
- Potter, Ron. "A New Order Tracking Method for Rotating Machinery." *Sound and Vibration*, September 1990.

Measurement data available with Order Analysis mode

Power Spec CH1	yes
Power Spec CH2	yes*
Linear Spec CH1	
Linear Spec CH2	
Time Channel 1	yes
Time Channel 2	yes*
Windowed Time Channel 1	
Windowed Time Channel 2	
Capture Channel 1	yes
Capture Channel 2	yes*
Frequency Response	
Coherence	
Cross Spectrum	
Orbit	yes*
Math Function	yes
Data Register	yes
Waterfall Register	yes

Composite Power CH1	yes
Composite Power CH2	yes
Order Track CH1	yes
Order Track CH2	yes*
RPM Profile	yes
Normalized Variance CH1	
Normalized Variance CH2	
Auto Correlation CH1	
Auto Correlation CH2	
Cross Correlation	
Histogram/Time CH1	
Histogram/Time CH2	
Probability Density Function CH1	
Probability Density Function CH2	
Cumulative Density Function CH1	
Cumulative Density Function CH2	

* Available only when analyzer is in two-channel mode

Measurement Data and the Order Analysis Mode

With this instrument mode, you have the following measurement data available:

- *Time* (channel 1 or channel 2; available only when order tracking is off). This is the most recent sampled time record on which the analyzer performed an FFT operation. The time record (and the FFT of this time record) is the basic building block for all subsequent measurement data (the time data appears in the time domain and thus looks like an oscilloscope trace). You can use time data to characterize an input signal in the time domain—however, you must set the analyzer to full span to avoid visual distortion of the time trace and undesired filtering (see explanation under "Measurement Data and the FFT Analysis Mode").

- *Capture* (channel 1 or channel 2). This is a time-domain trace that shows the contents of the analyzer's capture buffer. Using the capture buffer is a way to record input data and play it back for later analysis by making measurements from this captured data (think of the capture buffer as a small data acquisition system or instrumentation recorder). To learn more about capture, see chapter 20, "Capture Concepts."

- *Power Spectrum* (channel 1 or channel 2; available only when order tracking is off). This is derived from the linear spectrum data. To yield a power spectrum from the input data, the analyzer multiplies the linear spectum data by its complex conjugate and then calculates rms values for each point along the frequency spectrum. Power spectrum data thus provides good rms approximation for noise and other signals. Keep in mind that with power spectrum data, no phase information is available.

- *Orbit* (available only when order tracking is off). This is time-domain data that shows the amplitude of channel one versus the amplitude of channel 2. This produces results that are similar to *Lissajous patterns* on an oscilloscope. Orbit data is often used to analyze rotating machinery—typically, to measure shaft runout with two transducers, one at 90 degrees to the other. Orbit data is useful for balancing and understanding rotor dynamics.

- *Composite Power* (channel 1 or channel 2; available only when order tracking is on). This is the combined power for *all* orders—not just the orders you've chosen to track.

- Order Track (channel 1 or channel 2; available only when order tracking is on). This is the measure of total power in any order (an order is a harmonic of shaft rotation frequency) as a function of RPM. It's useful when determining how a particular order of rotation excites a system throughout its operating range.

- *RPM Profile* (available only when order tracking is on). An RPM profile is a record of the power spectra at successive, equally spaced RPM intervals measured from transducers on rotating machinery. This characterizes vibration (or acoustic output) of a rotating machine throughout its range of operating speeds. The RPM profile display is a three-dimensional plot (the horizontal axis is frequency, the vertical axis is RPM, and the third axis is time). Equally spaced along the vertical axis are normalized power spectrum plots. The vertical axis for each spectrum is power (volts squared).

- *Math Function*. Math functions let you perform a variety of operations on current (or stored) traces. A math function can contain current input data, a stored trace, or a constant. Math functions are specified with operands and operators in infix (standard algebraic) notation.

- *Data Register* (eight available). Data registers are provided exclusively for intermediate storage of trace data. For example, you can use a data register as temporary storage for a trace during an HP Instrument BASIC program. Or you can use the trace data as part of a math function by specifying this data register (D1 through D8) as an operand. Data registers are also used when you want to view a trace stored on disk—to do this, you simply recall the stored trace file from the appropriate disk and then load it into a data register; then you select this data register for display using the [**Meas Data**] menu. Keep in mind that *all information stored in data registers is volatile and will be lost when you turn off the analyzer*—make sure you've copied all important traces to disk (except the volatile RAM disk) beforehand.

- *Waterfall Register* (eight available). Waterfall registers are provided exclusively for intermediate storage of waterfall-formatted trace data. Waterfall registers are similar to data registers, with an important exception—you cannot use a waterfall register in a math function. Like data registers, you can use a watefall register to view a trace stored on disk—to do this, you simply recall the waterfall trace file from the appropriate disk and then load it into a waterfall register; then you select this register for display using the [**Meas Data**] menu. Keep in mind that *all information stored in waterfall registers is volatile and will be lost when you turn off the analyzer*—make sure you've copied all important waterfall traces to disk (except the volatile RAM disk) beforehand.

Special Considerations for Time Data

Time data is available in FFT Analysis, Order Analysis, Correlation Analysis, and Swept Sine modes. Windowed time data (showing the attenuation of the parts of a time record with window operations) is available for FFT Analysis and Correlation Analysis mode. For Histogram/Time mode, unfiltered time data is available (see chapter 13).

To interpret time data properly, you should know the following:

- Time traces are not calibrated, so they display only an approximate amplitude value. Still, they are very useful since they show input data before the analyzer filters the input data prior to FFT processing.
- Time traces have more resolution than frequency-domain traces. Although the analyzer has a maximum display resolution of 801 points (800 lines) and cannot display data with greater resolution, time data has a 1024-point resolution (unlike a maximum of 801 points for frequency-domain data). You can take advantage of this better resolution by using the axes scale markers (under the [**Scale**] hardkey) to expand a displayed time trace.
- If you set the analyzer to measure full span, the time data you see is the actual input time record. This is raw, unfiltered input data—the signal from which all subsequent measurements are based. You can use this input time record to verify that there is indeed a signal. You can also use the time record to help you manually set the analyzer's input range. For some signals, you may want to turn off the analyzer's anti-alias filters (under the [**Input**] menu) to prevent ringing or distortion of square waves or transients when viewed in the time domain. Make sure, however, that you turn on the anti-alias filter before making measurements in the frequency domain (the anti-alias filters are necessary to ensure a good transformation from the time domain to the frequency domain).
- If you set the instrument to measure a specific bandwidth (something less than full span), the time record shows the raw input data after filtering. This lets you see if there's energy within the selected span. For zoomed time record displays (start frequency not equal to zero), the displayed amplitude is approximately one-half the actual amplitude.
- If the analyzer is making averaged measurements, the most recent time record added to the average is the one displayed. The analyzer does not show a time waveform that is a cumulative average, since all averaging is done after the time data has been transformed to the frequency domain.
- Although the time data looks like an oscilloscope display, the analyzer is not a digital oscilloscope. The time record represents samples of a waveform. The samples have enough information to accurately reconstruct the input signal—but the human eye may not properly perform the reconstruction. In fact, for frequencies that are higher than about ten percent of the frequency span, there will be noticeable visible distortion.

12

Correlation Analysis

Introduction

Correlation is a time-domain measurement that reveals similarities within a signal, or similarities between two different signals. Correlation is related to a mathematical process called *convolution*. Autocorrelation reveals similarities within an input time record, while cross correlation reveals similarities between one input channel and another input channel.

Correlation measurements are useful for electronic, mechanical, and acoustics applications. You can use autocorrelation to extract periodic signals hidden by noise or use cross correlation to determine path delays or time delays.

Correlation Displays

Although correlation is a time-domain measurement, correlation displays are somewhat unusual in that they do not always show the entire time record of the input signal. Of the three weighting functions available, only the Uniform (– T/2, T/2) weighting function lets you view autocorrelation or cross correlation for the entire time record.

If you use the Zero Pad (– T/4, T/4) or Zero Pad (0, T/2) weighting functions, you can only see half the time record. Both of these weighting functions remove parts of the time record. For the Zero Pad (– T/4, T/4) function, the first quarter and the last quarter of the time record are removed. For the Zero Pad (0, T/2) function, the last half of the time record is removed. In both cases, parts of the time record are removed to avoid wrap-around error. To learn more, see "Wrap-Around Error."

Autocorrelation

The HP 35665A makes an autocorrelation measurement by performing an FFT on the input time record (transforming input data to the frequency domain), multiplying this data by its complex conjugate, and then performing an inverse FFT to bring the data back into the time domain. The convolution calculations can be done much faster this way, in contrast to comparable calculations done exclusively in the time domain.

The analyzer performs an autocorrelation approximation by multiplying the input time data by its time-shifted version and summing over all points. The result is plotted as a function of the time-shifted value. A peak on the autocorrelation trace indicates the location of a similar signal—multiple peaks indicate that the original signal is repeated, but delayed (or advanced) relative to the trigger signal. Periodic signals obscured by noise become visible in the autocorrelation display.

You can use autocorrelation to detect echoes in a signal. For random noise, an echo appears as an impulse—if there is more than one echo, you will multiple peaks on the autocorrelation trace. Keep in mind that an echo appears as an impulse only if the delayed signal has not been filtered. The impulse broadens as the original random noise signal is filtered—in fact, the width of the peak is inversely proportional to the bandwidth of the source signal.

To determine the time delay (in seconds) of an echo, you can move the marker to the peak of the echo. *Note that there is always a correlated peak at zero lag*—this peak marks the original excitation signal. Any other peaks let you know that the excitation signal also appeared at another time relative to the original signal. The amplitude value at the zero lag point is the total power in the time record.

You can use autocorrelation to enhance a periodic signal in the presence of uncorrelated noise. In this case, you can find the period of the enhanced signal by finding the distance (using the relative marker) between successive peaks on the autocorrelation trace. Keep in mind that the width of each peak is inversely proportional to the bandwidth of the signal. This is why random noise produces a very narrow peak. Other types of signals do not exhibit such sharp peaks. Correlated sine waves, for example, simply appear as cosine waves on the correlation trace.

Autocorrelation is a single-channel measurement. If you have the original signal on one channel and the delayed version on another, use cross correlation (cross correlation compares one channel to another).

Cross Correlation

The HP 35665A makes a cross correlation approximation by performing an FFT on both the channel 1 and the channel 2 input time records (transforming the input data to the frequency domain). The analyzer then multiplies the complex conjugate of channel 1 by channel 2 and then performs an inverse FFT to bring the data back into the time domain.

The analyzer approximates cross correlation by showing the similarity between two signals as a function of the time shift between them. The analyzer approximates cross correlation by multiplying one signal by a time-shifted version of another signal and summing over all points. The result is plotted as a function of the time shift value.

Cross correlation is useful for determining time delays between signals, one of which may be obscured by noise. It is often used for acoustics applications. For example, you can use cross-correlation to measure the speed of sound or to characterize echoes. Usually, one microphone is placed at a sound source and the other some distance away (the first microphone is usually connected to channel 1; the other to channel 2). The cross-correlation trace would show a peak at the time delay between the two microphones. Additional peaks might indicate echoes caused by reflected paths.

As with autocorrelation, you can use cross correlation to detect echoes. For random noise, an echo appears as an impulse—if there is more than one echo, you will multiple peaks on the autocorrelation trace. Keep in mind that an echo appears as an impulse only if the delayed signal has not been filtered. The implulse broadens as the original random noise signal is filtered—in fact, the width of the peak is inversely proportional to the bandwidth of the source signal.

To determine the time delay (in seconds) of an echo, you can move the marker to the peak of the echo. *Note that there is always a correlated peak at zero lag*—this peak marks the original excitation signal. Any other peaks let you know that the excitation signal also appeared at another time relative to the original signal. The amplitude value at the zero lag point is the total power in the time record.

Correlation Weighting Functions

Introduction

Because spectrum analyzers must use the FFT (Fast Fourier Transform) and not the Fourier Transform (which assumes an infinite block size), a peculiar type of error known as *wrap-around error* (also known as *circular convolution error*) occurs. To avoid wrap-around error, the analyzer lets you use *correlation weighting functions* (available under the [Window] menu). Wrap-around error causes amplitude inaccuracies and can seriously degrade autocorrelation or cross correlation measurements. You can avoid wrap-around error by using an appropriate correlation weighting function.

Wrap-Around Error

As we mentioned earlier, the HP 35665A converts time-domain data to the frequency domain as part of the correlation measurement. Any correlation measurement that calculates correlation in the frequency domain must compensate for an effect known as *wrap-around error.* Wrap-around error is sometimes called *circular convolution error.*

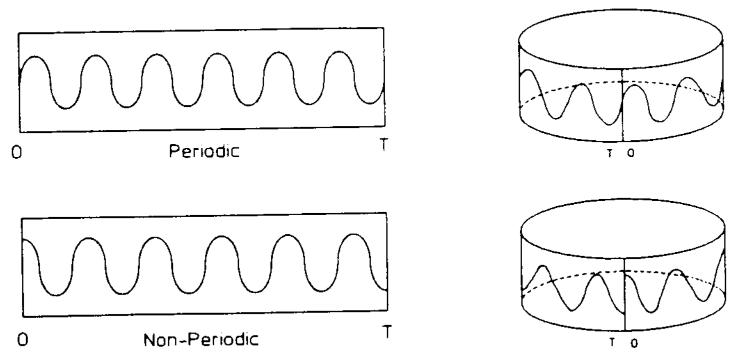
As we mentioned in chapter 2, "Measurement Basics," FFT spectrum analyzers sample the input signal and use an FFT algorithm to convert the input data from the time domain to the frequency domain. Each FFT measurement is based on a time record—the amount of time-domain data the analyzer needs to perform one FFT operation. The time record and its FFT are the building blocks the analyzer needs for all subsequent measurements.

To better understand wrap-around error, consider how the analyzer performs a correlation measurement:

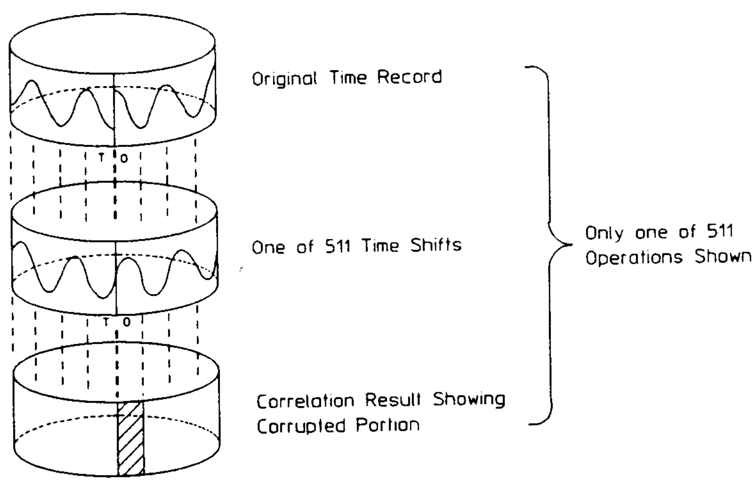
1. The analyzer weights the input time data.
2. It performs an FFT on both the channel 1 and the channel 2 input time records. This transforms the input data to the frequency domain, producing a linear spectrum.
3. For autocorrelation, the analyzer multiplies the linear spectrum for a single channel by its complex conjugate. For cross correlation, it multiplies the real part of channel 1 data by the complex conjugate of channel 2 data.
4. The analyzer performs an inverse FFT to bring the data back into the time domain.

The length of an individual time record is length "T" (usually indicated by the Greek letter τ). For both autocorrelation and cross correlation measurements, the analyzer takes this original time record, stores it, and compares it to successive time-shifted versions of this same input record. (Remember that autocorrelation is the multiplication of the input time data by its time-shifted version and summing over all points; cross correlation is the multiplication of one signal by a time-shifted version of another signal and summing over all points.)

For all FFT measurements, the analyzer assumes that the data within each time record is periodic. That is, if each time record were repeated, there would be no discontinuities. This means that the resulting time-domain waveform would be a realistic reconstruction of the original input signal. If you visualize a time record bent or wrapped around a circle, you can see that non-periodic signals would show a discontinuity where the strip is joined together (see the nearby illustration).



This type of discontinuity presents a special problem for correlation measurements. Imagine that you have two identical circular strips, one over the other. As you turn the bottom strip by one display point (essentially, what the analyzer does during correlation measurements) as it shifts the copied time record of the input data by one display point and then multiplies the two waveforms.



This shift is repeated, until the analyzer has completed all 511 time-shift operations. If the input signal was not periodic within the time record, parts of the resulting correlation waveform are corrupted because spurious parts of the time record are multiplied. The spurious part of the time record is the data to the left of the "seam" in the circular strip that represents the time record (see the nearby illustration).

The Weighting Functions

As we mentioned, each time record is stored and a copy is time-shifted many times to generate a correlation measurement. If the signal is not periodic within the time record, *the very first shift increment introduces wrap-around error.* The analyzer assumes that the signal is periodic.

The following are three correlation weighting functions used in the Correlation Analysis Mode:

- *Zero Pad (–T/4, T/4).* This function suppresses the first quarter and the last quarter of the time record, and passes the center part of the time record (the second and third quarters). *The trigger point (T_0) is at the center of the time record.* Use this function with random noise.

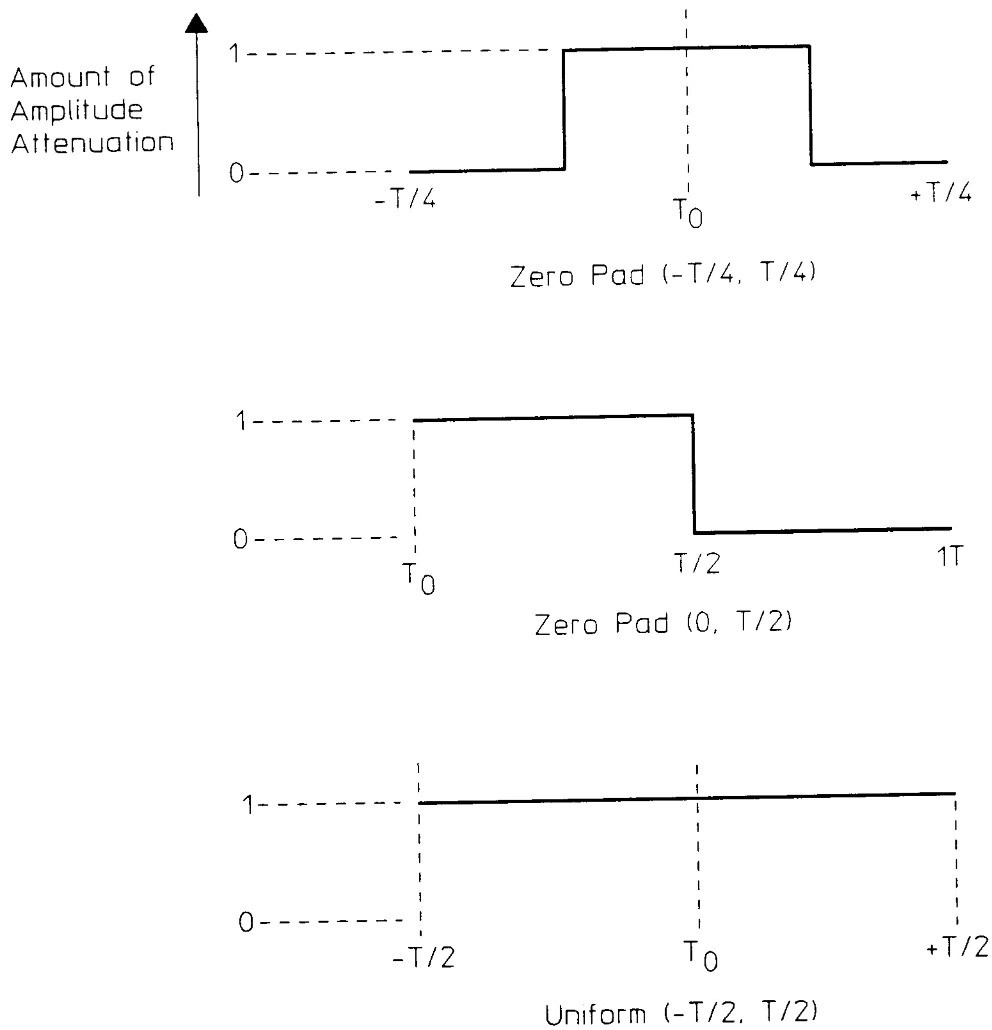
- *Zero Pad (0, T/2).* This function suppresses the last half of the time record, and passes only the first half. *The trigger point (T_0) is at the beginning of the time record.* Use this function with random noise.

- *Uniform (–T/2, T/2).* This function does not suppress any part of the time record. Because the Uniform function does not attenuate any part of the time record, is does not force an input signal to appear periodic in the time record. For best results with the Uniform function, you should use signal sources that are self-windowing, such a transients, bursts, and periodic waveforms. *The trigger point (T_0) is at the center of the time record.*

Note

It's important to know that correlation functions do not just attenuate parts of the time record—they also determine the relationship between the analyzer's trigger signal and the portion of the input signal sampled to make up a time record. Traditionally, correlation measurements are viewed with the trigger point (T_0) at the center of the display.

With the Zero Pad (–T/4, T/4) and Uniform (–T/2, T/2) functions, the trigger point is at the center of the display, since T_0 is at the *center* of the time record. With the Zero Pad (0, T/2) function, the trigger point is at the extreme left of the display, since the trigger point occurs at the *beginning* of the time record. If you need to change the trigger point, use pre- or post-trigger delay (to learn more about trigger delay, see chapter 5, "Triggering").



HP 35665A Correlation Weighting Functions

Special Considerations

For best results when making correlation measurements, consider the following:

- *Use ac coupling only.* Both autocorrelation and cross correlation measurements are disturbed by dc offsets in the input signal. Be sure to use select ac input coupling—you can do this from the [**Input**] menu.

- *Use appropriate triggering and trigger delays.* Triggering can be very important for some correlation measurements. For example, a consistent trigger signal lets you use vector averaging (vector averaging actually reduces noise; see chapter 6 to learn more about this type of averaging). Also, Pre- and post-trigger delays let you accurately position the input signal, with respect to time, within the time record—this is particularly important when using the –T/4 to T/4 weighting function and the 0 to T/2 function, as these function types attenuate parts of the sampled time record. Use *windowed time data* to align your signal and view the effects of using different trigger delays.

- *Use an appropriate weighting function and view the effects with Windowed Time trace.* There are three types of functions with correlation mode; Zero Pad (–T/4 to T/4), Zero Pad (0 to T/2), and Uniform (–T/2 to T/2). To view the results of these functions, use the *Windowed Time* trace. To learn more about correlation weighting functions, see "Correlation Weighting Functions" earlier in this chapter.

- *Use a random noise source for delay measurements.* Both autocorrelation and cross correlation measurements provide the ability to resolve time differences between waveforms that appear to be random.

- *Averaging may be useful.* Averaging is not necessary to make good correlation measurements. If you have a stable signal, you can use rms averaging; if the your signal is drifting, you can use rms exponential averaging. You can also use vector averaging to reduce noise, if you can provide a consistent trigger signal. To learn more about averaging, see chapter 6.

- *Waveforms on the correlation trace may not appear as they do on a time trace.* This is particularly noticeable when you are using correlation measurement to extract synchronous signals from noise. The enhanced signal shown on the correlation trace may be changed—for example, a correlated square wave appears as a triangle wave on the correlation trace. Other types of signals, such a sine waves, do not appear significantly distorted. The different shape of some waveforms is a direct result of the mathematical definition of correlation. *It's important to remember that the period of the waveform is preserved even if the correlation waveform looks different.* Thus you can identify correlated components even if their waveform shapes are not what you'd expected.

What is Correlation?

The *autocorrelation* function $R_{xx}(\tau)$ is a special time average defined by:

$$R_{xx}(\tau) = \lim_{T \to \infty} \frac{1}{T} \int_{T} x(t)\, x(t+\tau)\, dt$$

Thus, the autocorrelation function is founded by taking a signal, multiplying it by the same signal displaced τ units in time, and averaging the product over all time.

For the sake of simplicity and speed, most digital signal analyzers perform the correlation operation by taking advantage of its duality with the power spectrum; that is,

$$R_{xx}(\tau) \leftarrow\rightarrow G_{xx}(f)$$

Thus,

$$R_{xx}(\tau) = F^{-1}\left[G_{xx}(f)\right] = F^{-1}\left[S_x(f) \times S_x(f)^*\right]$$

The autocorrelation function always has a maximum at $\tau=0$ equal to the mean square value of x(t). If the signal x(t) is periodic, the correlation function is also periodic with the same period. Random noise, on the other hand, only correlates at $\tau=0$.

You can use the autocorrelation function to improve the signal-to-noise ratio of periodic signals. The random noise component concentrates near $\tau=0$ while the periodic component repeats itself with the same periodicity as the signal. Another thing to remember is that impulsive noises siuch as pulse trains, bearing ping, or gear chatter show up more distinctly in correlation or time record averaging than in a frequency-domain analysis.

Cross correlation is a measure of the similarity between two signals:

$$R_{yx}(\tau) = \lim_{T \to \infty} \frac{1}{T} \int_{T} y(t)\, x(t+\tau)\, dt$$

It is calculated the same way as the autocorrelation, using the dual relationship,

$$R_{yx}(\tau) \leftarrow\rightarrow G_{yx}(f)$$

Thus,

$$R_{yx}(\tau) = F^{-1}\left[S_y(f) \times S_x(f)^*\right]$$

Cross correlation indicates the similarity between two signals as a function of time shift (τ). One application for cross correlation is to determine time delays beween signals. These signals can be impulsive (radar or sonar application, for example) or broadband random noise such as that encountered in system stimulus response measurements (transmission path delays, room acoustics, airborne noise analysis, and noise source identification.

Measurement data available with Correlation Analysis mode

Power Spec CH1		Composite Power CH1	
Power Spec CH2		Composite Power CH2	
Linear Spec CH1		Order Track CH1	
Linear Spec CH2		Order Track CH2	
Time Channel 1	yes	RPM Profile	
Time Channel 2	yes*	Normalized Variance CH1	
Windowed Time Channel 1	yes	Normalized Variance CH2	
Windowed Time Channel 2	yes*	Auto Correlation CH1	yes
Capture Channel 1	yes	Auto Correlation CH2	yes*
Capture Channel 2	yes*	Cross Correlation	yes*
Frequency Response		Histogram/Time CH1	
Coherence		Histogram/Time CH2	
Cross Spectrum		Probability Density Function CH1	
Orbit		Probability Density Function CH2	
Math Function	yes	Cumulative Density Function CH1	
Data Register	yes	Cumulative Density Function CH2	
Waterfall Register	yes		

* Available only when analyzer is in two-channel mode

Measurement Data and the Correlation Analysis Mode

With Correlation Analysis mode, you have the following measurement data available:

- *Time* (channel 1 or channel 2). This is the most recent sampled time record on which the analyzer performed a correlation measurement. You can use time data to characterize an input signal in the time domain—however, you must set the analyzer to full span to avoid visual distortion of the time trace and undesired filtering (see related sidebar).

- *Windowed Time* (channel 1 or channel 2). This is similar to time data, but shows the time record after the analyzer has applied the correlation weighting function. Like the time trace, the windowed time trace represents the most recent sampled time record on which the analyzer performed the correlation measurement. Windowed time data lets you see the effects of the correlation weighting function on a particular input signal.

- *Capture* (channel 1 or channel 2). This is a time-domain trace that shows the contents of the analyzer's capture buffer. Using the capture buffer is a way to record input data and play it back for later analysis by making measurements from this captured data (think of the capture buffer as a small data acquisition system or instrumentation recorder). To learn more about capture, see chapter 20, "Capture Concepts."

- *Auto Correlation* (channel 1 or channel 2). This displays the similarity between a signal and a time-shifted version of itself. Auto correlation is approximated by multiplying the input signal by its time-shifted version and summing over all points. The result is plotted as a function of the time-shifted value.

- *Cross Correlation* (channel 1 or channel 2). This displays the similarity between two signals as a function of the time shift between them. Cross correlation is approximated by multiplying one signal by a time-shifted version of another signal and summing over all points. The result is plotted as a function of the time shift value. Cross correlation is most useful for determining time delays between signals, one of which may be obscured by noise.

- *Math Function.* Math functions let you perform a variety of operations on current (or stored) traces. A math function can contain current input data, a stored trace, or a constant. Math functions are specified with operands and operators in infix (standard algebraic) notation.

- *Data Register* (eight available). Data registers are provided exclusively for intermediate storage of trace data. For example, you can use a data register as temporary storage for a trace during an HP Instrument BASIC program. Or you can use the trace data as part of a math function by specifying this data register (D1 through D8) as an operand. Data registers are also used when you want to view a trace stored on disk—to do this, you simply recall the stored trace file from the appropriate disk and then load it into a data register; then you select this data register for display using the [**Meas Data**] menu. Keep in mind that *all information stored in data registers is volatile and will be lost when you turn off the analyzer*—make sure you've copied all important traces to disk (except the volatile RAM disk) beforehand.

- *Waterfall Register* (eight available). Waterfall registers are provided exclusively for intermediate storage of waterfall-formatted trace data. Waterfall registers are similar to data registers, with an important exception—you cannot use a waterfall register in a math function. Like data registers, you can use a waterfall register to view a trace stored on disk—to do this, you simply recall the waterfall trace file from the appropriate disk and then load it into a waterfall register; then you select this register for display using the [**Meas Data**] menu. Keep in mind that *all information stored in waterfall registers is volatile and will be lost when you turn off the analyzer*—make sure you've copied all important waterfall traces to disk (except the volatile RAM disk) beforehand.

Special Considerations for Time Data

Time data is available in FFT Analysis, Order Analysis, Correlation Analysis, and Swept Sine modes. Windowed time data (showing the attenuation of the parts of a time record with window operations) is available for FFT Analysis and Correlation Analysis mode. For Histogram/Time mode, unfiltered time data is available (see chapter 13).

To interpret time data properly, you should know the following:

- Time traces are not calibrated, so they display only an approximate amplitude value. Still, they are very useful since they show input data before the analyzer filters the input data prior to FFT processing.
- Time traces have more resolution than frequency-domain traces. Although the analyzer has a maximum display resolution of 801 points (800 lines) and cannot display data with greater resolution, time data has a 1024-point resolution (unlike a maximum of 801 points for frequency-domain data). You can take advantage of this better resolution by using the axes scale markers (under the [**Scale**] hardkey) to expand a displayed time trace.
- If you set the analyzer to measure full span, the time data you see is the actual input time record. This is raw, unfiltered input data—the signal from which all subsequent measurements are based. You can use this input time record to verify that there is indeed a signal. You can also use the time record to help you manually set the analyzer's input range. For some signals, you may want to turn off the analyzer's anti-alias filters (under the [**Input**] menu) to prevent ringing or distortion of square waves or transients when viewed in the time domain. Make sure, however, that you turn on the anti-alias filter before making measurements in the frequency domain (the anti-alias filters are necessary to ensure a good transformation from the time domain to the frequency domain).
- If you set the instrument to measure a specific bandwidth (something less than full span), the time record shows the raw input data after filtering. This lets you see if there's energy within the selected span. For zoomed time record displays (start frequency not equal to zero), the displayed amplitude is approximately one-half the actual amplitude.
- If the analyzer is making averaged measurements, the most recent time record added to the average is the one displayed. The analyzer does not show a time waveform that is a cumulative average, since all averaging is done after the time data has been transformed to the frequency domain.
- Although the time data looks like an oscilloscope display, the analyzer is not a digital oscilloscope. The time record represents samples of a waveform. The samples have enough information to accurately reconstruct the input signal—but the human eye may not properly perform the reconstruction. In fact, for frequencies that are higher than about ten percent of the frequency span, there will be noticeable visible distortion.

13

Histogram/Time Measurements

Introduction

Histogram Measurements

Histogram measurements show how the amplitude of the input signal is distributed between its maximum and minimum value during a series of averaged measurements. This gives you an indication of how often a particular signal stays at a particular amplitude level. Histograms are often used to determine the statistical properties of noise and to monitor the performance of electromechanical positioning systems.

Histogram results are viewed as number of average counts (Y-axis) versus input amplitude (X-axis). Thus they are often said to be in the *amplitude domain*. This is different than the time-domain or frequency-domain displays used with the analyzer's other measurements.

Unlike other averaged measurements, histogram measurements *do not combine the results of each average*. Instead, each average count is characterized individually and then added to the amplitude distribution shown on the histogram display. The number of averages for a histogram simply determines how many time records are characterized. Keep in mind that the accuracy of a histogram depends on the frequency span, time record length, and the number of averages you've specified.

Hint: *For an optimal histogram measurement, set the histogram length (in points) equal to the number of bins squared.*

Time Measurements

The Histogram/Time mode also lets you view an input signal in the time domain—in effect, an "oscilloscope" mode. Because time data in the Histogram/Time mode is not processed through anti-aliasing filters or bandwidth-limiting filters, it is called "unfiltered time" data.

Although time data is available in other instrument modes, *unfiltered* time data available with the Histogram/Time offers these advantages:

- *No anti-alias filtering.* Because the Histogram/Time mode does not use the FFT algorithm to transform data to the frequency domain, the analyzer does not use its anti-aliasing filters. Anti-aliasing filters can cause ringing for some types of input signals, such as square waves, transients, or pulses. You should know that for the analyzer's other instrument modes, you can turn off these anti-aliasing filters from the [**Input**] menu.

- *No digital filtering to limit the bandwidth to the current frequency span.* With the unfiltered time data, you can look at time data for *any* frequency span, without losing information (for most other instrument modes, time data is available but anything less than full-span limits the frequency response of the time data).

- *Better amplitude accuracy, regardless of span.* Two things contribute to better amplitude accuracy with unfiltered time data—no anti-aliasing filters and the fact that the time data is not filtered to restrict its bandwidth. So with unfiltered time data, you can look at time data for *any* frequency span, without losing information (for other instrument modes, time data is available but anything less than full-span limits the frequency response of the time data).

You should know that there may be some visual aliasing of the unfiltered time data. The best way to view time data is start with the largest span, then work your way down to provide the resolution you need. To control the span size, you can change the time record length from the [**Freq**] menu.

Like time data with the analyzer's other instrument modes, you can use the analyzer's time parameter markers to measure all or part of this unfiltered time data, with regard to the following:

- Overshoot.
- Rise time.
- Settling time.
- Delay time.

Note

Although Histogram/Time mode is the only instrument mode that does not use anti-aliasing filtering or digital filtering, you can disable the anti-alias filtering in *any* instrument mode (this is done from the input setup menu).

Display Resolution and Histogram/Time Mode

Measurement data available with the Histogram/Time mode has 1024 points of resolution, in contrast to a maximum of 801 points available with most of the analyzer's other instrument modes. Because the analyzer can display only up to 801 points, however, histogram and time traces do not *appear* to have better resolution. However, you can expand the displayed trace to view histogram or time traces with increased resolution. To learn more about scale expansion, see chapter 16, "Display Formats."

Measurement data available with Histogram/Time mode

Power Spec CH1	
Power Spec CH2	
Linear Spec CH1	
Linear Spec CH2	
Unfiltered Time Channel 1	yes
Unfiltered Time Channel 2	yes *
Windowed Time Channel 1	
Windowed Time Channel 2	
Capture Channel 1	yes
Capture Channel 2	yes *
Frequency Response	
Coherence	
Cross Spectrum	
Orbit	
Math Function	yes
Data Register	yes
Waterfall Register	yes

Composite Power CH1	
Composite Power CH2	
Order Track CH1	
Order Track CH2	
RPM Profile	
Normalized Variance CH1	
Normalized Variance CH2	
Auto Correlation CH1	
Auto Correlation CH2	
Cross Correlation	
Histogram CH1	yes
Histogram CH2	yes *
Probability Density Function CH1	yes
Probability Density Function CH2	yes *
Cumulative Density Function CH1	yes
Cumulative Density Function CH2	yes *

* Available only when analyzer is in two-channel mode

Measurement Data and the Histogram/Time Mode

With this instrument mode, you have the following measurement data available:

- *Histogram* (channel 1 or channel 2). Histogram data shows how the amplitude of the input signal is distributed between its maximum and minimum value during a series of averaged measurements. This gives you an indication of how often a particular signal stays at a particular amplitude level. Histograms are often used to determine the statistical properties of noise and to monitor the performance of electromechanical positioning systems.

- *Probability Density Function* (channel 1 or channel 2). This is similar to histogram data, except that the Probability Density Function (PDF) is a normalized version of the histogram. PDF is a statistical measure of the probability that a specific level occurred. The analyzer calculates PDF by multiplying the number of averages by the number of points in the time record (1024, maximum) and dividing the histogram by this value (the probability of an input signal falling between two display points is equal to the integral of the curve between these two points). To learn more about display points, see chapter 3, "Spectrum Analyzer Review."

- *Cumulative Density Function* (channel 1 or channel 2). The analyzer calculates the Cumulative Density Function (CDF) by integrating PDF results. The CDF shows the probability that a level equal (or less) than a specific level occurred. You can also use CDF to check for dc offsets with symmetrical signals.

- *Unfiltered Time* (channel 1 or channel 2). This is a time-domain trace that shows the current input time record without anti-alias filtering or bandwidth-limiting filtering. Otherwise, unfiltered time data is similar to time data available with the analyzer's other instrument mode—and you can use time data to characterize an input signal in the time domain. Keep in mind that you may have to adjust the frequency span (by specifying the time record length) to avoid visual distortion of the time trace.

- *Capture* (channel 1 or channel 2). This is a time-domain trace that shows the contents of the analyzer's capture buffer. Using the capture buffer is a way to record input data and play it back for later analysis by making measurements from this captured data (think of the capture buffer as a small data acquisition system or instrumentation recorder). To learn more about capture, see chapter 20, "Capture Concepts."

- *Math Function*. Math functions let you perform a variety of operations on current (or stored) traces. A math function can contain current input data, a stored trace, or a constant. Math functions are specified with operands and operators in infix (standard algebraic) notation.

- *Data Register* (eight available). Data registers are provided exclusively for intermediate storage of trace data. For example, you can use a data register as temporary storage for a trace during an HP Instrument BASIC program. Or you can use the trace data as part of a math function by specifying this data register (D1 through D8) as an operand. Data registers are also used when you want to view a trace stored on disk—to do this, you simply recall the stored trace file from the appropriate disk and then load it into a data register; then you select this data register for display using the [**Meas Data**] menu. Keep in mind that *all information stored in data registers is volatile and will be lost when you turn off the analyzer*—make sure you've copied all important traces to disk (except the volatile RAM disk) beforehand.

- *Waterfall Register* (eight available). Waterfall registers are provided exclusively for intermediate storage of waterfall-formatted trace data. Waterfall registers are similar to data registers, with an important exception—you cannot use a waterfall register in a math function. Like data registers, you can use a watefall register to view a trace stored on disk—to do this, you simply recall the waterfall trace file from the appropriate disk and then load it into a waterfall register; then you select this register for display using the [**Meas Data**] menu. Keep in mind that *all information stored in waterfall registers is volatile and will be lost when you turn off the analyzer*—make sure you've copied all important waterfall traces to disk (except the volatile RAM disk) beforehand.

14

Marker Concepts

Introduction

The analyzer has a [**Marker**] hardkey and a [**Marker Fctn**] hardkey. These keys let you use a variety of marker features. This chapter shows you some of the way that you can interpret your measurements using the softkeys in the [**Marker**] menu. If you want to learn more about a specific softkey in this menu, see the *HP 35665A Operator's Reference*.

The Absolute Marker

The absolute marker appears as a small diamond on the analyzer's screen. As you turn the knob, the absolute marker moves. The marker readouts (just above each trace box) indicate the X-axis and Y-axis values of the current marker position.This marker is called the absolute marker (or main marker) because its values reflect the absolute X-axis and Y-axis values. This is different than the *relative marker.*

Markers are extremely useful. You can use them to search for peaks and to find specific values. You can also use the relative marker to find relative values between two points.

Relative Marker

The relative marker is a second marker whose X and Y positions are relative to a marker reference. The marker reference appears as a small square.

When the relative marker is on, the X-axis and Y-axis values indicated are those relative to the position of the marker reference (the square), not absolute values. The marker labels change from X and Y to Xr and Yr.

To move the marker reference to a specific (absolute location), use the [REFERENCE X ENTRY] and [REFERENCE Y ENTRY] softkeys under [REFERENCE SETUP]. To move the marker reference to the position of the main marker, press [REFERENCE TO MARKER].

Using the Markers

Changing from Absolute to Relative Marker

To toggle a marker between absolute and relative, use the [MKR VAL **ABS** REL] softkey.

Turning on and off Markers

Each trace has its own marker. To turn on a marker for a particular trace, first make sure the trace is active (use the [**Active Trace**]. The marker options you select apply only to the marker on the active trace.

Marker Coupling

Marker coupling means that the markers for both traces move together. Marker coupling is quite useful. For example, if you display frequency response magnitude on the upper trace, and phase on the lower, you can use marker coupling to read both magnitude and phase at each frequency bin along the X-axis.

Moving the Marker to a Specific Location

You can move the marker to a specific location using the [MARKER X ENTRY] softkey.

Setting a Marker Reference

The [REFERENCE TO MARKER] menu lets you establish a reference point for any X-axis or Y-axis value.

- [REFERENCE TO MARKER] turns on the relative marker and moves it to the main marker location.
- [REFERENCE X ENTRY] moves the marker reference to a specific X-axis value.
- [REFERENCE Y ENTRY] moves the marker reference to a specific Y-axis value.

Searching for a Peak

Press [MARKER TO PEAK] to move the marker to the largest amplitude on the displayed portion of the trace. Keep in mind that this moves the marker to the peak only for the trace that's currently active (but the marker on the inactive trace will also move if marker coupling is on).

Once moved, the marker remains at the new X-axis location until you do one of the following things:

- Turn the knob (with no entry window displayed).
- Press another marker-search key.
- Enter a new X-axis location (using [MARKER X ENTRY]).

Note

The analyzer can automatically move the marker to the highest peak each time the active trace is updated. To enable this feature, turn on peak tracking.

Next Peak Right and Next Peak Left

[NEXT PEAK RIGHT] moves the marker to the right, to the next local maximum on the trace. The [NEXT PEAK LEFT] softkey moves the marker to the left, to the next local maximum.

The marker moves to the next right peak or next left peak only on the trace that's active (but the marker on the inactive trace will also move if marker coupling is on).

Note

The absolute marker does not move outside the displayed X-axis boundaries. It can, however, move outside the displayed Y-axis boundaries (though it doesn't disappear from the display entirely).

The relative marker, however, is not displayed if it is anchored to an X-axis or Y-axis point outside the displayed X-axis or Y-axis.

Peak Tracking

The [PEAK TRK **ON** OFF] softkey toggles the peak track feature for the active trace. When peak tracking is on, the analyzer continuously moves the marker to the peak value on the trace.

You can turn on peak tracking for Trace A, Trace B, or both traces. If you turn on peak tracking for both traces, each marker follows the peak for its respective trace (unless marker coupling is on).

Note

Because marker coupling takes priority over peak tracking, the marker will not track the peak value for the inactive trace if both marker coupling and peak tracking are turned on.

15

Marker Function Concepts

Introduction

The analyzer has a [**Marker**] hardkey and a [**Marker Fctn**] hardkey. This chapter shows you some of the way that you can interpret your measurements using the softkeys in the [**Marker Fctn**] menu. It also provides a general overview of each marker function. To learn more about individual softkeys in these menus, see the *HP 35665A Operator's Reference*.

The [**Marker Fctn**] hardkey lets you use the following specialized markers:

- Harmonic markers.
- Band markers.
- Sideband markers.
- Waterfall markers.
- Time Parameters.
- Gain and phase margins.
- Resonant frequency and damping.

Marker Function Availability

Not all marker functions are available with all types of measurement data or with every instrument mode. For example, gain/phase margins are valid only with frequency response data—therefore, the [GAIN/PHAS MARGINS] is ghosted unless you've displayed frequency response data.

Marker Functions and their Availability

	FFT analysis	Octave Analysis	Order Analysis	Swept Sine	Correlation Analysis	Histogram /Time
Harmonic Marker	yes		yes	yes		
Band Marker	yes	yes	yes	yes		
Sideband Marker	yes		yes	yes		
Waterfall Markers	yes*	yes*	yes*		yes*	yes*
Time Parameters	yes†		yes†	yes†	yes†	yes†
Gain/Phase Margins	yes‡			yes‡		
Frequency and Damping	yes‡			yes‡		

* To use waterfall markers, you must select the waterfall display format and make trace B active.
† Available only for time data
‡ Available only for frequency response data

Harmonic Markers

Harmonic markers let you calculate total harmonic distortion (THD) or harmonic power for any frequency-domain display. You can specify the fundamental frequency and the analyzer indicates the harmonics for that particular fundamental.

You can specify the number of harmonics that you want the analyzer to use for both THD and harmonic power calculations. The actual number of harmonics identified depends on the fundamental frequency and the analyzer's bandwidth. Higher fundamental frequencies have fewer harmonics displayed, because it takes fewer harmonics to reach the top end of the analyzer's frequency range.

For THD measurements, the analyzer displays THD as a percentage of the amplitude at the fundamental frequency. Keep in mind that the analyzer calculates THD by comparing the energy of the fundamental to the energy of the harmonics. Noise and other signals at other points along the frequency spectrum are not taken into account (unless they happen to occur at the fundamental frequency or at the harmonics). This is different than the THD+N (total harmonic distortion plus noise) measurements made by traditional distortion analyzers.

For harmonic power measurements, the analyzer measures the absolute value of the identified harmonics of the fundamental frequency. Again, remember that noise and other signals at other points along the frequency spectrum are not taken into account (unless they happen to occur at a harmonic frequency).

Note

If the trace coordinate is dB magnitude, the analyzer displays harmonic power in dBVrms. For other trace coordinates, the analyzer displays harmonic power in Vrms^2.

Band Markers

You can use the analyzer's band markers to define a frequency band and then calculate the power or rms square root power within this band. You can specify the band by entering a start and stop frequency. Alternatively, you can specify a center frequency and a band span value.

Band power is the total power within the specified frequency band. The value is displayed in the lower left corner of the trace box. Band power calculations are expressed using the currently-selected vertical units.

Note

If the trace coordinate is dB magnitude, the analyzer displays harmonic power in dBVrms. For other trace coordinates, the analyzer displays harmonic power in Vrms ^ 2.

Sideband Markers

Sideband markers let you determine sidebands (and sideband power) for a particular carrier frequency. You can do this by specifying the carrier frequency, the number of sidebands, and the sideband increment (the frequency difference between the sidebands and the carrier frequency). The sideband power value represents the rms summation of all marked sidebands.

Keep in mind that the carrier frequency you specify does not have to be within the current frequency span. Also, the actual number of sidebands identified depends on the carrier frequency and the analyzer's bandwidth.

Note

If the trace coordinate is dB magnitude, the analyzer displays harmonic power in dBVrms. For other trace coordinates, the analyzer displays harmonic power in Vrms ^ 2.

Waterfall Markers

Waterfall markers are specialized markers that you can only use with the analyzer's waterfall display. A waterfall display is a continuous map that shows multiple measurements. Each waterfall trace represents an individual time record—the latest time record is at the bottom of the trace; the newest at the top. To learn more about waterfall displays and the types of measurements you can view with the waterfall format, see chapter 16, "Display Concepts."

An important concept here is that there are two distinct ways to examine data with the waterfall display—by selecting a trace or by selecting a slice. A *trace* is simply one time record that you can select from the waterfall display (a trace could be in the frequency domain or in the time domain). A *slice* is a special type of trace that the analyzer constructs from the current waterfull buffer. We'll explain more about slices and slice markers in a moment.

Note

Waterfall markers are valid only for waterfall displays and when trace B is the active trace. If you are making a measurement, you must first pause the analyzer to use the waterfall markers.

You can use the waterfall markers to do the following things:

- Scroll up or scroll down in the waterfall trace buffer.
- Select a trace or slice from the waterfall.
- Display a trace or slice in the upper trace.
- Save a trace or slice to a data register.
- Set up the waterfall display.

Scrolling in the Waterfall Display

More traces may be stored in a waterfall than are currently displayed. You can use the [SCROLL UP] softkey to display a group of traces below (older than) the currently displayed traces. You can use the [SCROLL DOWN] softkey to display a group of traces abover (newer than) the currently displayed traces.

Selecting a Trace or Slice

One of the advantages of the waterfall display is that you can view measurement data over time. When you find a trace you want to examine, you can select a trace (or slice) from the waterfall display and save that to a data register. From there, you can examine the data, use it for math operations, or save it to disk.

A *slice* is a vertical line through the collection of waterfall traces at the same X-axis value. The number of points in the slice is equal to the number of traces in the waterfall buffer. Each point is the amplitude for the corresponding trace.

If the number of traces is so large that the data register cannot store all the data, the slice begins with the selected trace and ends with the last trace that fits into the register.

Displaying Traces or Slices

To display a selected trace or slice, you must first save it to a data register and then recall that data register. To make this easier, there is a [SAVE AND DISP DATA] softkey. When you press [SAVE AND DISP DATA], the trace (or slice) you selected previously is automatically saved and displayed.

To learn more about using waterfall markers, see the *HP 35665A Operator's Reference*.

Time Parameters

Introduction

Time parameters let you calculate values for the following:

- Overshoot.
- Rise time.
- Settling time.
- Delay time.

The analyzer calculates these values between a specified start and stop time. You can specify these times with the [START TIME] and [STOP TIME] softkeys. Then use the numeric entry keys or the knob to move the start and stop time markers to their respective locations.

Overshoot

Overshoot is the the maximum value by which a step response exceeds its steady-state level. The analyzer uses only the data between the start time and stop time markers in the computation.

Rise Time

Rise time is the time required for a step response to rise from 10% to 90% of its steady-state level. The analyzer uses only the data between the start time and stop time markers in the computation.

Note

The analyzer uses the lowest steady-state value and approximate highest steady-state value between start time and stop time. The start time should be at least five bins before the transition for the computation to be accurate.

Settling Time

Settling time is the time required for a step response to reach steady-state level and stay within +/− 2 percent of that level. The analyzer uses only the data between the start time and stop time markers in the computation. Settling time is measured from the start time marker.

Delay Time

Delay time is the time required for a step response to reach 50 percent of its steady-state value. The analyzer uses only the data between the start time and stop time markers in the computation. The delay time is measured from the start time marker.

Gain/Phase Margins

The gain/phase margin markers let you specify a start and stop frequency, then compute and display gain/phase margins and crossovers. These markers are valid only for frequency response data.

Gain margin is defined as the magnitude level (in dB) when the phase crosses below – 180 degrees. A value in the range of +/– 6 dB indicates the possibility of an unstable system.

Phase Margin is defined as 180 degrees minus the absolute value of the phase angle when the gain is equal to 0 dB or 1. A value less than +/– 10 indicates the possibility of an unstable system.

If the gain or phase crossover occurs between measured data points, the actual crossover is linearly interpolated. The analyzer begins searching for zero gain and phase at the start and continues until it finds the first gain and phase crossovers. If the analyzer finds no crossovers before reaching the stop marker, it displays "undefined" instead of the value.

The gain/phase margin values appear at the bottom of the trace box. The gain/phase crossovers appear as solid vertical lines in the trace box. The values of the crossovers also appear in the mini-state for the marker functions.

Frequency and Damping

The frequency and damping markers let you calculate and display the resonant frequency and the damping ratio for frequency response data. The analyzer uses a 1 degree of freedom curve fitter on data between the start and stop markers. For accurate results, the marker band should cover at least the 3 dB bandwidth.

Frequency and damping are respresented as a single coefficient. The analyzer calculates Frequency and damping using the following equations:

frequency = I
damping = R / sqrt (R^2 + I^2)

where:
R = real part of the comples pole pair corresponding to the resonance
I = imaginary part of the complex pole pair corresponding to the resonance

The algorithm for frequency and damping computes a conjugate pole pair, implicitly assuming an underdamped resonance. If the frequency response behaves as a critically or overdamped system between markers, the algorithm will return meaningless values. The damping is always less than 1. Frequency and damping can only be computed for complex frequency-domain data. It cannot be computed for Nyquist trace coordinates.

The damping computation for the HP 35665A is different than that for the HP 3562/3563; the computed frequency is the same for both analyzers. To convert HP 35665A damping to HP 3562/3563 damping, use the following equation:

a =freq / sqrt (1 − b^2)

where:
a = HP 3562/3563 damping
b = HP 35665A damping

16

Display Formats

Introduction

You can use the analyzer's [**Disp Format**] menu to view measurement data in the following formats:

- Single.
- Upper/Lower.
- Front/Back.
- Measurement State.
- Bode Diagram.
- Waterfall.

Not all display formats are appropriate for all instrument modes. Those formats that are inappropriate for a particular mode are unavailable when that instrument mode is active.

In addition, you can reduce the amount of information on the screen if you use the following display format features:

- Grid On Off
- Blank Annotation
- Blank Display
- Trace Title
- Default title on/off

Display Format Overview

The following overview describes all display formats except the waterfall format. To learn about waterfall displays, see "Waterfall Display Format" later in this chatper.

To learn more about individual softkeys in these menus, see the *HP 35665A Operator's Reference*.

Single Trace Format

This displays only the currently active trace using a single, full-height trace box. All trace annotation applies to the active trace.

Upper/Lower Trace Format

This displays both traces using two half-height trace boxes. Trace A is displayed in the upper box and trace B in the lower box. The annotation for each trace box applies for the trace displayed in the box. The annotation for the active trace is in a plain font; the annotation for the inactive trace is in a ghosted font.

Front/Back Trace Format

This displays both traces using one full-height trace box. The active trace is drawn with a solid line, the inactive trace with a dotted line. Annotation for the active trace is in the plain font; annotation for the inactive trace is in a ghosted font.

Measurement State

The measurement state shows the analyzer's current configuration—how you've set up the analyzer. Use this display and one of the plot or print softkeys to document the instrument setup for a particular measurement. You can also use this display while you are setting up a measurement. The analyzer updates the display when you change parameter settings.

Input State

This displays the analyzer's current input configuration—how you've set up the input channels and tachometer input. Use this display and one of the plot or print softkeys to document the input setup for a particular measurement. You can also use this display while you are setting up a measurement. The analyzer updates the display when you change input settings.

Bode Diagram

This displays a Bode diagram. For a transfer function, a Bode diagram is a plot of log gain and phase versus log frequency. For rotating machinery measurements, a Bode diagram is a plot of component amplitude and phase versus linear running speed. When you select the Bode diagram display format, the following changes are made in the display:

- The measurement data for traces A and B is changed to frequency response.
- The trace coordinate for trace A is changed to dB magnitude.
- The trace coordinate for trace B is changed to phase.
- The X-axis scale is changed to log.
- Markers are enabled and coupled.

You can change the X-axis back to linear spacing from the [**Trace Coord**] menu.

Trace Grid

You can turn on or off the overlay grid (graticule) for the active trace. Keep in mind that if you turn off a trace grid, it does not appear if you print or plot to an external printer/plotter.

Trace Title

You can assign a title for the active trace. Use the numeric keypad and the alpha keys to enter an appropriate name (up to 14 characters long). The title appears above the upper left corner of the trace box. Entering a trace title is useful for labeling results to be plotted.

You can also toggle between the default title and a trace label that you created. The default title is the name of the measurement data displayed in the trace.

Note The trace title *displayed* is the one kept when you save a trace. For example, if you create a trace title, but decide to use the default title instead, the default title is the title saved, not the trace title you created.

Blanking Trace Annotation

Each trace box is surrounded by fields that define the trace within that box. These fields are collectively referred to as "trace annotation." If necessary, you can blank (turn off) the trace annotation. When trace annotation is turned off, amplitude and frequency information is not displayed on the screen and it is not printed or plotted.

Note

You must preset the analyzer to turn on blanked annotation from the front panel. You must send DISP:ANN:FREQ ON over the HP-IB to turn on blanked annotation without presetting.

Blanking the Display

You can also blank (turn off) all information on the analyzer's screen except the softkey labels. When the screen is blanked, the **DISPLAY BLANKING ON** message replaces all other information. Only this message is plotted or printed.

Note

You must preset the analyzer to turn on a blanked screen from the front panel. You must send DISP:ENAB ON over the HP-IB to turn on a blanked screen without presetting.

Waterfall Display Format

Introduction

The waterfall format provides a small upper trace box and a larger lower trace box. The lower trace box displays trace B in a waterfall or map. The traces are scrolled down the display (the newest trace is at the top). The upper trace box displays trace A.

Trace A and trace B are independent in a waterfall display. You can assign any available measurement data to either trace. This is different from some other analyzers that assign the same measurement data for both traces. A quick way to get a waterfall display of current measurement data is to press the [WATERFALL ACT TRACE] softkey

The softkeys controlling the behavior of the waterfall display are accessed by pressing [WATERFALL SETUP]. Special waterfall markers are available under the **[Marker Fctn]** hardkey. You can use these markers to select a trace or slice from the lower trace box to be displayed the upper trace box.

Note The waterfall display format is not available for swept sine measurements. If you select the swept sine instrument mode when the current display selection is waterfall, the display format changes to single.

Waterfall Setup

You can set up waterfall displays with the [WATERFALL SETUP] sofktey in the [**Disp Format**] menu. You can select the following waterfall display characteristics:

- Specify the Z-axis range.
- Specify the trace height.
- Turn hidden lines on or off.
- Suppress the baseline.
- Turn on waterfall skew and select the skew angle.

Note You specify the number of traces stored for a waterfall by pressing [WATERFALL STEPS] under [Trigger], [ARM SETUP].

Z-Axis Range

You can specify the range of complete traces you want displayed on the waterfall. This range varies with the type of data you are displaying.

You can specify the Z-axis range in one of the following units:

- Number of counts
- Number of averages
- Seconds
- rpm

Waterfall Trace Height

You can specify the height of each trace displayed in the waterfall display area as a percentage of trace box height. The limits are 1 to 100 percent; the default value is 39 percent.

The top (most recent) trace in the waterfall uses the top "trace height" percent of the trace box. The base of the top trace is indicated by a line at the left edge of the display. The bases of the other traces displayed are spread evenly in the rest of the trace box.

For example, if you specify a trace height of 50 percent, each trace will be half the height of the trace box. If you display two traces, the traces will each take half of the trace box with no overlap. If you display more than two traces, the traces will overlap.

Note

If you set the trace height less than 29 percent or greater than 75 percent, some of the waterfall annotation is not displayed.

Hidden Line

You can turn on and off the *hidden line* in the waterfall display. When you remove the hidden line is removed, the analyzer displays all parts of every trace. This can clutter the display if you are displaying many traces. When hidden line removal is on, the analyzer does not display portions of traces that lie *behind* previous traces.

Baseline Suppression

You can specify a percentage of the waterfall traces to be suppressed in the display. The analyzer does not display this percentage of the bottom of each trace. Baseline suppression is useful for removing noise floor clutter from the display. You can set baseline suppression from 0 to 100 percent. The default value is 0 percent.

For example, if you set baseline suppress to 10 percent, the lower ten percent of each trace is not displayed. If the Y-axis scale is from – 100 dB to 0 dB, only amplitudes greater than – 90 db are displayed. If you set baseline suppress to 0 percent, the analyzer displays the full trace.

Special Considerations for Waterfall Displays with Trigger Arming

- For rpm step arm or time step arm, the spacing of the measurements may be different than specified for rpm step size or time step size. For runup or rundown measurements, the analyzer may not be able to process measurements as fast as the specified step size required. If this happens, the spacing between traces will vary, as the "missing" traces are not available to display when the measurement is paused or complete.

- Because of overlapping traces, there may appear to be more traces displayed than you specified. You can specify a trace height as a percentage of the trace box height. For example, if you specify a trace height of 50 percent, each trace will be half the height of the trace box. If you display only two traces, the traces will each take half of the trace box with no overlap. If you display more than two traces, the traces will overlap. While the measurement is running, there may be parts of more than this number of traces on the screen. When the measurement is paused, the partial overlapping traces are not displayed.

17

Scale Concepts

Introduction

You can use the [**Scale**] hardkey to select an appropriate scale and units for the active trace. This chapter gives you a brief overview of this menu. To learn more about individual softkeys in this menu, see the *HP 35665A Operator's Reference*.

Note

Some scale-related items, such as Y units and the choice of linear or logarithmic X-axis, are selected from the [**Trace Coord**] menu. To learn more about trace coordinates, see chapter 3, "Getting Comfortable with the HP 35665A Dynamic Signal Analyzer."

You can select the following items from the [**Scale**] menu:

- Y-axis autoscale.
- Y-axis top reference.
- Y-axis center reference.
- Y-axis bottom reference.
- Y-axis input range tracking.
- Y-axis per division.
- Match X-scale.
- Match Y-scale.
- Axes scale markers.

Note

The arrow keys and the knob are especially useful in the Scale menu, since they let you quickly change the vertical scaling.

General Scale Behavior

Autoscaling

Autoscaling lets the analyzer automatically select a vertical scale that best fits the active trace. The analyzer adjusts the scale, if necessary, each time the display is updated. If you prefer, you can select the scale manually using the other softkeys in the [Scale] menu.

Keep in mind that autoscaling can affect a waterfall display. If the scale changes, the analyzer clears the waterfall display and displays the next traces using the new scale. This affects only the display, not the measurement. The cleared traces are still kept in the waterfall buffer.

Scale References

If you don't want to use autoscaling, you can select top, center, or bottom scale references. The analyzer keeps these references regardless of your choice of y-units/division.

Here's how scale references work. If you're viewing frequency response data on the active trace, setting a top reference of 20 dB means that the top of the line in the display graticule is always at +20 dB. If you change the Y per div value, the top of the scale remains at +20 dB and the bottom of the scale changes accordingly.

Input Range Tracking

You can use input range tracking to reference the scale to the input range currently in use. This behavior varies according to the trace coordinate you've selected:

- For linear magnitude traces, the bottom reference always stays at zero. The Y per div is adjusted so the top of the scale is greater than or equal to the current input range.

- For logarithmic magnitude and dB magnitude traces, the top reference always stays at the current input range.

- For real and imaginary traces, the center reference always stays at zero. The Y per div is adjusted so the top of the scale is greater than or equal to the current input range.

- Phase traces do not use input range tracking. Input range tracking is turned off during an autoscale procedure or when you change the Y per div for real, imaginary, or linear magnitude traces.

Note

Input range tracking is not available when you display frequency response, coherence, or math functions.

Adjusting the Vertical Scale

Adjusting the Units/Scale Division

You can adjust the units/scale division by using the [Y PER DIV (DECADES)] softkey. For linear Y-axis, this specifies the number of units per vertical scale division. For a log Y-axis, this specifies the number of decades displayed.

When you select a new scale spacing, the currently active reference (top reference, center reference, or bottom reference) is held the same and the rest of the scale adjusted around this level. By the way, the reference softkey with a box around it is the currently-active reference.

Note

To choose Y units or to specify either a linear or logarithmic X-axis, use the appropriate sofkeys under the [**Trace Coord**] menu. To learn more about trace coordinates, see chapter 3, "Getting Comfortable with the HP 35665A Dynamic Signal Analyzer." To learn more about specific softkeys, see the *HP 35665A Operator's Reference*.

Matching X and Y Scales

You can use the [MATCH X SCALE] softkey to set the X-axis scale of the inactive trace to the active trace. The [MATCH Y SCALE] softkey is similar, but sets the Y-axis scales to the same scaling.

This lets you easily set both traces to identical X-axis or Y-axis scales. This makes it easier to compare data on both traces, particularly when you have the upper/lower or front/back display formats selected.

Axes Scale Markers

The HP 35665A offers a feature similar to the one on the HP 3562A/B analyzer—axes scaling. This lets you expand the displayed trace to examine it more closely. Using the axes scale markers, you can:

- Specify X-axis or Y-axis scaling.
- Move markers using the knob or numeric entry keys.
- Return to a full-span display.
- Expand the band identified by the markers to fill the display.
- Specify which of the markers should hold its position and which should move.

Note

It's important to remember that axes scaling only expands the displayed data—it does not give you better measurement resolution. If you need more resolution to interpret a measurement, either increase the display resolution or reduce the analyzer's frequency span—then make another measurement.

To learn more about the axes scale markers, see the *HP 35665A Operator's Reference*.

18

Source Overview

Introduction

The analyzer's source can supply a number of different waveforms. For example, you can select any of the following waveforms with the FFT Analysis mode (keep in mind that not all source waveforms are available with each instrument mode):

- Random noise.
- Burst random.
- Periodic chirp.
- Burst chirp.
- Pink Noise
- Fixed sine.
- Arbitrary source (optional).

Source Impedance

The analyzer's source impedance is very low (less than 5 ohms) and is designed to be operated into nearly any type of load. If your device-under-test required a specific source impedance, be sure to place an appropriate resistor in series with the analyzer's source output. For example, if your device-under-test requires a source impedance of approximately 600 ohms, you can insert a 590 ohm resistor in series with the analyzer's source output.

Caution Although the analyzer's source has protection circuitry, do not apply more than ± 42 volts (peak) to the center conductor of the source connector (referenced to ground).

Source Waveforms available with each instrument mode

	FFT analysis	Octave Analysis	Order Analysis	Swept Sine	Correlation Analysis	Histogram/ Time
Random Noise	yes	yes	yes		yes	yes
Burst Random	yes				yes	yes
Periodic Chirp	yes				yes	yes
Burst Chirp	yes				yes	yes
Pink Noise	yes	yes	yes		yes	yes
Fixed Sine	yes	yes	yes		yes	yes
Arbitrary Source	yes *				yes *	yes *
Swept Sine				yes ‡		

* Available only with Option 1D4
‡ Available only with Option 1D2

Source Type Overview

The following section provides a brief overview of the available source types. If you need more information, see the *HP 35665A Operator's Reference.*

Random Noise

Random noise yields a fast, linear estimate of the system under test. Because it is not periodic in the time record, random noise requires windowing (usually the Hanning window).

Burst Random

For Burst Random noise, the source outputs a random noise waveform during the specified percentage of the time record and nothing during the remainder of the record. You can specify the percentage of the time record used (the default value is 50 percent).

Because the burst random output is periodic, it's best to use the Uniform window when making measurements using this waveform. Also, source triggering is recommended for burst source outputs.

Periodic Chirp

Periodic chirp is a fast sine sweep across the current frequency span that repeats with the same period as the time record. The effect of the periodic chirp is similar to the random noise waveform, but the chirp has a much lower peak-to-rms ratio and its spectrum is much flatter than a noise spectrum. And because the chirp waveform is the same in every time record, averaging may not be required for measurements using the periodic chirp.

Because the chirp output is periodic, it's best to use the Uniform window when making measurements using this waveform.

Burst Chirp

A chirp is a fast sine sweep across the current frequency span that repeats with the same period as the time record. Burst chirp allows you to specify the percentage of the time record that the source is active (the default is 50 percent).

Because the burst chirp output is periodic, it's best to use the Uniform window when making measurements using this waveform. Also, source triggering is recommended for burst source outputs.

Pink Noise

Pink noise is similar to random noise, except that the spectral density is inversely proportional to frequency. This means that the amplitude rolls off at 3 dB/octave.

Pink noise is used for octave measurements (a typical use for pink noise is microphone calibration). Because the octave bands are wider at higher frequencies, the random noise levels are considerably greater at higher frequencies. Pink noise compensates for the wider high-frequency bands.

Fixed Sine

The Fixed Sine waveform is simply a sine wave at a frequency which you specify.

Swept Sine

The Swept Sine source is available only with Option 1D2.

The Swept Sine source is activated automatically when you select the Swept Sine instrument mode. The source sweeps to let you make frequency response measurements. To learn more about swept sine measurements, see chapter 9, "Swept Sine Measurements."

You can also specify a *ramp rate* to determine how fast the swept source changes when you start, stop, pause, and continue a swept sine measurement. The source also ramps to a different level if you change the source level during a swept sine measurement or if the autolevel algorithm changes the level.

In Swept Sine mode, you can use an *autolevel* feature that lets the analyzer adjust the source output level to keep the amplitude of one input channel within a specified range. When autolevel is off, the source has a constant amplitude (level) at all measurement points.

When autolevel is on, the amplitude at the measurement frequency is monitored. At each measurement point, the source amplitude is adjusted until the reference input amplitude is within a specified tolerance band around the reference level. You can specify the reference channel, reference level, reference tolerance, the maximum source output, and the maximum input level in the autolevel setup menu.

Arbitrary Source

The Arbitrary Source is available only with Option 1D4.

The Arbitrary Source lets you drive the source output with the contents of a data register (the register must contain time domain data). The analyzer scales the data so that its peak voltage corresponds to the current source level in Vpk. Then the analyzer outputs the scaled signal to the source connector. You can specify which data register should be used by pressing [ARB SRC SETUP] and then [DATA REG Dx].

Because the arbitrary source waveform is only one time record long, you may have to use the repeat feature. When repeat is on, the analyzer outputs data to the source connector continuously, without interruption. When repeat is off, the source behavior is affected by trigger mode.

In free run trigger mode, source output is continuous, just as it is with repeat on. For any other trigger mode, the source begins its output only when a trigger occurs and shuts off after all the data in the register has been output. This happens each time a trigger occurs.

General Source Behavior

The analyzer remembers a separate set of source parameters for each instrument mode. The exceptions are source on/off and source level. The level setting remains unchanged when you change instrument modes. However, when you change modes, the source is turned off (except when switching to Swept Sine mode).

For swept sine measurements, the menu includes ramp rate and autoleveling parameters. Fixed sine is the only source output type for swept sine measurements. The analyzer's source is not available with the order analysis instrument mode (Option 1D0).

Caution

When you turn on the analyzer's power (and when you turn off power), a brief pulse may appear at the source output connector. Do not cycle power if you have sensitive test devices connected to the analyzer's source.

When you first turn on the analyzer (or press [**Preset**]), the source selected will be fixed sine, and it will be turned off. Unlike most source parameters, source on/off is global for all instrument modes (except swept sine and order analysis). This means that if you turn on the source, then change the instrument mode, the source remains on. The analyzer does not remember a different source on/off state for each instrument mode.

Note

If you turn off the source and turn it on again, the output level will automatically return to the level you set previously—even if you've selected a different source waveform.

Note

When you start a measurement with capture on, the analyzer turns off the source. The toggled highlight of the [SOURCE ON OFF] softkey does not change. The source status is indicated by the SRC LED above the trace. When the measurement is complete, the analyzer returns the source to its original on/off state.

Setting the Source Level

The source level that you specify applies to all waveforms. You can specify peak volts, true rms volts, peak dBV, or rms dBV (the absolute limits of the source level are 0 Vpk to 5 Vpk for most source types). If you don't specify a new output level, the output levels remain at the level you set previously.

For random noise or periodic chirp, the level you set is the total wideband level—in other words, the summation of these waveforms measured at full span. If you're using smaller frequency spans, not all of this energy will appear in the measurement because some of the waveform's power will be outside the selected span.

If you specify the source level with one type of unit (Vrms, for example) but want to see what this same level would be using a different unit (such as dBVpk), there's a convenient way to do this. For example:

1. Enter a level of 1 Vrms (press [LEVEL], [1], [Vrms]).
2. Press [.] on the number keypad.
3. Press the softkey for the unit you prefer (for example, [dBVpk]). The source level is now displayed as 3.01 dBVpk (for a fixed sine source output).

The Global Nature of Source Level

Unlike most source parameters, the source level is global for all instrument modes. This means that if you specify a source level, then change the instrument mode, the source level remains the same. The analyzer does not remember a different source level for each instrument mode.

The analyzer also keeps track of the units you specified for the level. For example, if you set the level in V rms for random noise, the analyzer maintains the rms level when you change to a different source output type. If you set the level in V peak, the analyzer maintains the peak level.

Crest Factors

Because the crest factors for different source output types are different, the peak value changes when the analyzer maintains the rms value. The source output level is limited to +/− 5 V peak. If the new rms value would require a peak value outside this range, the analyzer sets the rms value to the maximum possible.

For example, the crest factor for random noise is approximately 4.4. The crest factor for fixed sine is 1.414. If you set a source level of 1 V rms for random noise and then change the output type to fixed sine, the analyzer maintains the 1 V rms level. The peak voltage required for 1 V rms is 4.4 V for random noise and 1.414 V for fixed sine.

19

Time Capture Concepts

Introduction

Time capture allows you to record real-time data containing frequencies up to 51.2 kHz (2 channel) or 102.4 kHz (1 channel). The analyzer stores the data to a dedicated capture buffer. The *capture* trace is a time-domain trace that shows the contents of this capture buffer.

Using the capture buffer is a way to record input data and play it back for later analysis by making measurements from this captured data. You could think of the capture buffer as a small data acquisition system or instrumentation recorder.

The capture feature is available with all instrument modes. To use the capture feature, you use the functions in the [CAPTURE SETUP] menu.

Capture Overview

As we mentioned, time capture allows you to record real-time data containing frequencies up to 51.2 kHz (2 channel) or 102.4 kHz (1 channel). The analyzer stores the data to a capture buffer. After the capture is complete, you can save the captured data in the buffer to disk.

The size of this capture buffer depends on the amount of memory available in your analyzer. If necessary, you can free up more memory by removing waterfall displays, HP Instrument BASIC programs, and RAM disk from memory. To check the memory allocation, press [**System Utility**] and press [MEMORY USAGE].

You must set up the following parameters before capturing data:

- Instrument Mode (you can change this when you analyze the data).
- Number of channels.
- Capture length.
- Tach data on/off.
- Frequency settings.
- Triggering.
- Source (if using internal source).
- Input setup.

You can setup or change the following parameters after capturing data:

- Instrument Mode.
- Averaging.
- Windowing.
- Analysis region.

If there is not enough memory for the specified time length, the analyzer allocates as much memory as possible and displays an error message. When the capture is complete, the analyzer toggles to [CAPTURE **ON**OFF].

Capturing Data

Starting the Capture

To start the capture, you press [START CAPTURE]. The analyzer then begins collecting data from the inputs and store it in the time capture buffer. If memory was not previously allocated for the time capture buffer, it is allocated now.

Aborting Capture

To stop the capture, you press [STOP CAPTURE]. This stops the the time capture process. Data already in the time capture buffer is retained. The actual amount of data will be less than the specified capture length.

Capture Setup Considerations

Capture Length

You can specify the length of the time capture buffer. You can specify seconds, number of records, or number of points. The limits for capture length vary depending on such things as the amount of memory available, the number of channels, and the frequency span. Memory is not allocated until you press [START CAPTURE] or [ALLOCATE CAPTURE].

Allocating Capture

You can force the immediate memory allocation of the time capture buffer based on the defined capture length, number of channels, and the frequency span. The analyzer asks you for confirmation before it performs the allocation. If there is not enough memory for the specified time length, the analyzer allocates as much memory as possible and displays an error message.

Removing Capture

You can also deallocate the memory reserved for time capture. Again, the analyzer asks you for confirmation before it deallocates the memory. (This is equivalent to deleting the capture buffer from [MEMORY USAGE] menu under the [**System Utility**] hardkey.)

Tachometer Parameters

You can specify if you want to include tachometer data in the capture buffer. However, you must first set up the tachometer parameters before capturing the data.

Analysis Region

You can specify the portion of the time capture buffer data to be used for a measurement. You can also specify a start time and stop time for each channel individually. The start time and stop time are referenced to the beginning of the capture buffer. If the data is triggered, the times are referenced to the trigger point.

Capture Header

The capture header identifies captured data. This displays the following information:

- Length (in seconds and number of records).
- Number of channels.
- Frequency span.
- Center frequency.
- File size (in bytes).
- Whether tachometer data was used.

Tachometer Data

If you used tachometer data, the header also indicates the following:

- RPM at beginning of capture data.
- RPM at end of capture data.
- Maximum order of capture data.

Using Captured Data

Before you use results from captured data, you must toggle to [CAPTURE **ON** OFF] in the [**Inst Mode**] menu.

Toggling the [CAPTURE ON OFF] softkey lets you specify whether measurement data comes from the time capture buffer or the input channels. When capture is on, the analyzer takes data from the time capture buffer. When capture is off, the analyzer takes data from the input channels.

Note

When you start a measurement with capture on, the analyzer turns off the source. When the measurement is complete, the analyzer returns the source to its original on/off state.

When you start a time capture, the analyzer automatically toggles to [CAPTURE **ON** OFF] when the capture is complete.

Note

The message "End of CAPTURE data" appears when the analyzer reaches the end of the time capture data. This message also appears if there is no time capture data or the *capture length* is less than the *record length*.

20

Analysis Operations

Analysis Overview

You can use the analyzer's [**Analys**] menu to do the following:

- Set up math operations (defining functions and constants)
- Perform limit testing
- Perform synthesis operations
- Perform curve fit operations
- Edit data registers

Math Operations

Introduction

Using constants and functions, you can perform a variety of operations on current (or stored) traces. You can use math to modify the results of a measurement—for example, to compensate for a fixed gain (or loss) in a system- or device-under-test. You can also use a math operation to modify a trace—for example, to invert a trace. And you can use math to add, subtract, multiply, or divide traces with other traces.

Special Considerations

It's not difficult to use math operations, but there are some things you should know before building your own math functions:

- A math function can contain the current input data, a stored trace, or a constant.
- Math functions are specified by entering the definition with operands and operators in infix (standard algebraic) notation.
- Constants must be defined as real numbers.
- To view the results of a math operation, press [**Trace Data**] and use the appropriate softkeys to call up the results for a particular math function.
- To exit any math menu without affecting any function of constant definitions, simply press any hardkey.

Using Logarithmic Values with Math Operations

Keep in mind that the analyzer performs math operations in linear units, not logarithmic units. This is true regardless of trace coordinate. All math operations occur *before* the analyzer converts measurement data to the display units. This is important to keep in mind when you need to add (or subtract) units with logarithmic values—for example, if you need to compensate for a gain or loss in your measurement system.

Suppose you want to add 3 dB to your measurement results. You can't build a math function that simply adds 3 dB since math operations are done with linear units. Instead, you must take a trace and multiply it by the antilog of the offset (in dB) that you want. This converts the 3 dB to linear units—and you can express this offset as 10 to the x/20, where x is the dB offset.

So if you want to add 3 dB, you must multiply a trace by 1.413. If you want to subtract 3 dB, you can divide the trace by 1.413 (or conversely, you could multiply the trace by 0.707).

Limit Testing

Introduction

A limit test is a line (or set of lines) that you create to check the performance of a signal source or a device-under-test. When limit testing is on, the analyzer compares a current measurement or a stored trace to the limit you've selected.

A limit appears as a single line (upper or lower limit) or two lines (upper and lower limit). If a trace exceeds the boundaries of these lines, the limit test fails. Limit testing is useful for go/no go checking since a limit test quickly tells you if your device-under-test passes or fails a particular limit test.

You can build a limit line—an upper limit, a lower limit, or set of both upper and lower limits—in several different ways:

- By using the knob (or numeric entry) to arbitrarily construct a limit line.
- By saving a trace, converting it to a limit, and shifting this newly-created limit up or down to form an upper or lower limit.
- Via HP-IB.

To learn how to build limits, see the *HP 35665A Quick Start Guide*.

Synthesis and Curve Fit

Curve Fit and Synthesis are sophisticated features that are explained in chapters 21 and 22. The following paragraphs provide only a brief description of these features.

Synthesis

The synthesis feature lets you generate frequency-response traces based on a pole/zero model. You can use synthesis to estimate the frequency response of theoretical circuit. You can also specify a system gain factor, time delay, and frequency scaling when modeling complex systems.

Curve Fit

The curve-fit feature lets you find a mathematical model that closely approximates the frequency response data obtained from measured or synthesized data. The HP 35665A lets you determine up to 20 poles/zeroes to characterize a frequency response trace. After performing a curve fit, you can view the results in tables, showing either pole/zero, polynomial, or pole-residue formats.

Editing Data Registers

Data editing allows you to change the trace stored in a data register. You can define a line segment by specifying the start X and stop X positions, then entering the start Y and stop Y values. The analyzer connects the two points with a straight line. You can insert as many lines as you need.

You must modify start Y or stop Y. If you do not change either Y-value, the analyzer does not change the data between the two points. You can then move start X and stop X to define a new band.

You can edit a data register to do the following:

- *Modify a trace to remove an objectionable component.* For example, you can "zero out" part of a trace by placing defining a line segment from point to point. You can do this for any type of trace.
- *Modify a waveform to use as an arbitrary source.* For example, you can remove the lower half of a sine wave (in the time-domain) by drawing a line segment between the negetive crossover points of the sine wave.

21

Synthesis

Introduction

This chapter shows you how to use the analyzer's synthesis feature and also provides some basic concepts. The tasks in this chapter are common to many synthesis procedures. However, if you need information about a specific softkey, see the *HP 35665A Operator's Reference* or use the analyzer's online help facility.

The synthesis feature lets you generate frequency-response traces based on an H(s) model. You can use synthesis to estimate the frequency response of theoretical circuit. You can also specify a system gain factor, time delay, and frequency scaling when modeling complex systems. Although you can enter data into a synthesis table at any time, you must be in either the FFT Analysis mode or the Swept Sine mode to do a synthesis calculation.

What is Synthesis?

Synthesis lets you create a trace based on the transfer function of a system. Using the synthesis menus under the [**Analys**] hardkey, you can enter parameters from your transfer function into a synthesis table. Then the analyzer can use this synthesis table—with either the FFT Analysis mode or the Swept Sine mode—to create a frequency response trace.

Each synthesis table can have up to 20 poles (10 conjugate pairs) *and* 20 zeros (10 conjugate pairs). You can enter synthesis parameters in one of the following data formats:

- Pole-zero
- Pole-residue (partial fraction)
- Polynomial

Once you create a synthesis table, you can store it as you would a trace or a math function. You can store a table into the analyzer's NVRAM, RAM, or internal disk. You can also store a table to an external disk via the HP-IB.

The Synthesis Calculation

The synthesized trace is displayed with the X-axis in Hertz. Consequently, for H(s), the parameters (such as pole positions) must be in Hertz. Since the X-axis is in Hertz, the synthesis equation for H(s) is as follows:

$$H(s) \mid s = jf$$

where: f is frequency in Hertz

Remember that H(s) transfer functions are frequently expressed in radians. The analyzer expects parameters from the transfer function to be in Hertz. Sometimes you may be starting with an H(s) expressed in radians.

Note

If you modify H(s) to express it in Hertz, you must adjust the gain of H(s) to keep the dc gain constant. To do this for pole-zero or polynomial transfer functions, adjust the gain by a factor of $(2\pi)^{(n-m)}$, where *n* is the order of the numerator, and *m* is the order of the denominator. To do this for pole-residue transfer functions, you must adjust the gain for each term in the partial-fraction-expansion. For a simple pole, this means dividing the residue by 2π. For a complex-conjugate pole pair, this means dividing the residue by $(2\pi)^2$.

X-Axis Spacing

Synthesis data spacing for the x-axis can be either linear or logarithmic. Use the X-AXIS LIN LOG softkey under the [SYNTHESIS SETUP] menu to set the mode before performing the synthesis.

Start and Stop Frequencies

The analyzer uses the frequency-span set up for measurements to determine the synthesis start and stop frequencies.

Other Parameters

There are a few additional parameters that you can enter into the synthesis calculation. These are gain factor, time delay, and frequency scale. You can enter these parameters under the [SYNTHESIS SETUP] menu.

Gain Factor

This is an optional parameter that you can use to set the gain of the synthesized frequency response function. The default value for the gain factor is 1.0.

Time Delay

If necessary, you can enter a time delay for a synthesis calculation. The default is 0 seconds, which has no effect on the synthesis. If you enter a non-zero time delay (t), the H(s) synthesis includes multiplication by:

$$e^{-j2\pi ft}$$

where: f is frequency in Hertz
t is time delay

Frequency Scale

This scales the frequency axis (the X-axis) by:

$$\frac{f}{\textit{freq scale}}$$

where: f is frequency in Hertz
freq scale is the scale frequency

Thus, if you want the frequency axis to be in radians, enter a scale frequency of $1/(2\pi)$. Then enter parameters for H(s) in terms of radians (even though the unit keys still say mHz, kHz, and Hz). The frequency axis is always labeled Hz (Hertz); however, the x-axis cursor should now be interpreted as radians. Units *are not* displayed by frequency scale in the synthesis table because this is a frequency scaling operation, and hence has units such as Hz/Hz, Hz/kHz, or Hz/mHz.

Task 1a: Entering Data in Pole-Zero Format

1. Press [**Analys**],

   press [SYNTHESIS],

   press [CONVERT TABLE].

Before you can enter data, you must specify the table format you're going to use, *if you do not already have an existing table in that format.*

2. Press [CONVRT TO POLE ZERO].

   If the table is already in the pole-zero format, skip this step.

This converts the current format to the pole-zero format.

For pole-zero data, the synthesis table is split into two columns: the left side for poles and the right side for zeros. Poles (or zeros) can be either real or complex conjugate pairs.

Conjugate pairs, real poles, and real zeroes all occupy one line. Often, real poles are called "simple poles" and real zeroes are called "simple zeros." A $\pm j$ distinguishes a complex conjugate pair from a simple pole or zero.

Now press [EDIT TABLE].

You can enter up to 20 poles and 20 zeros (the order is not critical). You can enter any combination of real or complex pole-zero data. However, if every line contains a conjugate pair, you can only enter a maximum of 10 lines.

Note that the requirement of conjugate pairs means that only Hermitian symmetric pole-zero data is supported.

3. Using the numeric keypad, enter the first pole in the **POLES** column. Use the appropriate softkeys to select the correct suffix for each pole.

   An example of an entry would be 2 + j5 (this means 2 ± j5).

   To add zeroes, use the knob to move the highlighted bar to the **ZEROS** column. Enter the zeroes the same way you entered the poles.

   If you make a mistake, move the highlighted bar over the entry you want to change. Then use the [CHANGE VALUE] softkey to modify your entry.

   You can also use the [ADD VALUE], [DELETE VALUE], and [UNDELETE VALUE] softkeys.

   To clear the table, press [CLEAR TABLE] and then press [CONFIRM CLEAR].

4. Press [**Freq**].

   Using the appropriate softkeys, set the start and stop frequencies for the synthesis.

   Once you set these, you don't have to set them for subsequent synthesis calculations.

5. *This step is optional.*

   Press [SYNTHESIS SETUP].

   Using the appropriate softkeys, enter parameters for gain factor, time delay, or frequency scale. You can also toggle the X-axis between linear and logarthmic.

   For the example pole-zero equation shown after this task, you would need to enter 1, then press [EXP], enter 6, and press [ENTER], enter. The gain then appears as `1e+06` at the bottom of the synthesis table.

6. You can now run the synthesis.
   To do so, go to Task 2.

The synthesis trace is stored in a dedicated register (data register D8 is the default register, though you can change this). You can run the synthesis without saving the trace to disk. However, the data register is overwritten the next time you create a new synthesis trace.

To save your synthesis table to disk, press [**Save / Recall**], press [SAVE MORE], and press [SAVE SNTH TABLE]. Then using the alpha keys (or an external keyboard), enter an appropriate filename.

To save your synthesis trace to disk, press [**Save / Recall**], press [SAVE DATA], press [SAVE TRACE], and press [INTO FILE]. Then using the alpha keys (or an external keyboard), enter an appropriate filename.

Hints on Entering Pole-Zero Data:

To express your transfer function in pole-zero format, use the following formula:

$$H(s) = K \frac{(s - Z_1)(s - Z_2)(s - Z_2^*)..(s - Z_n)}{(s - P_1)(s - P_2)(s - P_3)(s - P_3^*)..(s - P_m)}$$

where:

- * is the conjugate
- K is the system Gain
- P is the pole position
- Z is the zero position

This formula is the basis for interpreting pole-zero data in the synthesis table. Pole and zero positions are in Hertz. During synthesis, the analyzer replaces s with jf (where f is frequency in Hertz).

Pole-Zero Example

This is an example of a pole-zero data that you enter into a synthesis table:

$$H(s) = \frac{1 \times 10^6 (s - 600)}{(s + 250)(s + 500 - j2000)(s + 500 + j2000)}$$

where: pole and zero terms are in Hz
(thus, frequency scale = 1.0 will be used)

To enter this pole-zero example, follow the steps in Task 1a. When you are done, your table should look like the one in figure 21-1.

Disk Util Def Disk: Internal

POLES 3	SYNTH	ZEROS 1
-250		600
-500 ±j 2e+3		
TIME DELAY = 0 s		GAIN = 1e+06
FREQUENCY SCALE = 1		

Figure 21-1. Synthesis Table for Pole-Zero Example

Task 1b: Entering Data in Pole-Residue (Partial Fraction) Format

1. Press [**Analys**],

   press [SYNTHESIS],

   press [CONVERT TABLE].

Before you can enter data, you must specify the table format you're going to use, *if you do not already have an existing table in that format.*

2. Press [CONVRT TO POLE RESD].

This converts the current format to the pole-residue (partial fraction) format.

   If the table is already in the pole-residue, skip this step.

   Now press [EDIT TABLE].

For pole-residue data, the synthesis table is split into two columns: the left side for poles and the right side for residues. A residue is another name for the numerator constant for each pole term in the partial-fraction expansion.

Poles can be simple (real) or entered as a complex conjugate pair. A pole and its residue are always on the same line in the synthesis table (unlike pole-zero tables, where order doesn't matter). A $\pm j$ distinguishes a complex conjugate pole-pair from a simple pole.

Since a complex conjugate pole-pair corresponds to two terms in a partial-fraction expansion, the residue is also interpreted as being a complex-conjugate pair.

If you enter all complex-conjugate pairs, you can only enter up to 10 lines. However, you can enter up to 20 lines of simple poles and simple residues.

3. If you don't need to use the **1s ^ 0** residue term, press [CLEAR TABLE].

   This term appears when you convert an empty table (interpreted as H(s) =1) to the pole-residue format.

The **1s ^ 0** residue term simply adds a "1" to the synthesis equation. An s ^ n term is called a Laurent term. Laurent terms cannot be edited on the synthesis table.

4. Using the numeric keypad, enter the first pole to the **POLE** column. Use the appropriate softkeys to select the correct suffix for each pole.

   To add residues, use the knob to move the highlighted bar to the **RESIDUES** column.

If you make a mistake, move the highlighted bar over the entry you want to change. Then use the [CHANGE VALUE] softkey to modify your entry.

You can also use the [ADD VALUE], [DELETE VALUE], and [UNDELETE VALUE] softkeys.

To clear the table, press [CLEAR TABLE].

5. Press [**Freq**].

   Using the appropriate softkeys, set the start and stop frequencies for the synthesis.

Once you set these, you don't have to set them for subsequent synthesis calculations.

6. *This step is optional.*

   Press [SYNTHESIS SETUP].

   Using the appropriate softkeys, enter parameters for gain factor, time delay, or frequency scale. You can also toggle the X-axis between linear and logarithmic.

   For the example pole-residue equation shown after this task, you would need to enter 1, then press [EXP], enter 6, and press [ENTER], enter. The gain then appears as `1e+06` at the bottom of the synthesis table.

7. You can now run the synthesis. To do so, go to Task 2.

   The synthesis trace is stored in a dedicated register (data register D8 is the default register, though you can change this). You can run the synthesis without saving the trace to disk. However, the data register is overwritten the next time you create a new synthesis trace.

   To save your synthesis table to disk, press [**Save / Recall**], press [SAVE MORE], and press [SAVE SNTH TABLE]. Then using the alpha keys (or an external keyboard), enter an appropriate filename.

   To save your synthesis trace to disk, press [**Save / Recall**], press [SAVE DATA], press [SAVE TRACE], and press [INTO FILE]. Then using the alpha keys (or an external keyboard), enter an appropriate filename.

Hints on entering Pole-Residue Data

Overview

To express your transfer function in pole-residue format, use the following formula:

$$H(s) = K\,[c_1 s^0 + c_2 s^1 + \,.\,.\, +$$

$$\frac{A_1}{(s - P_1)} + \frac{A_2}{(s - P_2)} + \frac{A_2^*}{(s - P_2^*)} + \frac{A_3}{(s - P_3)} + \frac{A_3^*}{(s - P_3^*)} +$$

$$\frac{A_4}{(s - P_3)^2} + \frac{A_4^*}{(s - P_3^*)^2} + \,.\,.\,]$$

where:

- * is the conjugate
- K is the system Gain
- P are the pole positions
- A are the residues (a constant, possibly complex)
- c terms caused by numerator order ≥ denominator order

This formula is the basis for interpreting pole-residue data in the synthesis table. Pole positions are in Hertz. During synthesis, the analyzer replaces s with jf (where f is frequency in Hertz). If the same pole (real or conjugate) is entered more than once, the pole term is surrounded by parenthesis and its order is indicated by ^n, where n is the multiplicity. There are no c terms if the numerator order is less than the denominator order.

Note

For pole-residue, the values in the synthesis table must be identical in order to be identified as a multiple order pole. Thus, if you convert the synthesis table from polynomial to pole-residue format, then add a pole that appears to be the same, it probably won't be identified as the same pole due to rounding errors in the conversion.

Pole-Residue Example

This is an example of a pole-residue data that you enter into a synthesis table. It's the same data as in the pole-zero example, but written as pole-residue data.

You can perform a partial-fraction expansion on the following pole-zero equation to yield a corresponding pole residue equation:

Pole-zero equation:

$$H(s) = \frac{1 \times 10^6 (s - 600)}{(s + 250)(s + 500 - j2000)(s + 500 + j2000)}$$

Pole-residue equation:

$$H(s) = \frac{-209.24}{s + 250} + \frac{104.61 - j263.08}{s + 500 - j2000} + \frac{104.61 + j263.08}{s + 500 + j2000}$$

If you haven't cleared the pole-zero data that you entered in Task 1a, you can generate a pole-residue table by pressing [CONVERT TABLE] and [CONVRT TO POLE RESD].

To enter this pole-residue example, follow the steps in Task 1b. When you are done, your table should look like the one in figure 21-2.

Analysis

POLES 3	SYNTH RESIDUES
-250	-209.24
-500 ±j 2e+3	104.61 ±j -263.08
TIME DELAY = 0 s FREQUENCY SCALE = 1	GAIN = 1

Figure 21-2. Synthesis Table for Pole-Residue Example

Reconstructing Simple Poles

To reconstruct the pole-residue equation from the synthesis table, you must associate poles with their residues. Poles and their residues are always on the same line in the synthesis table.

The equation for a simple residue over a simple pole is:

$$\frac{\text{(Residue)}}{s - \text{(Simple Pole)}}$$

where: (residue) is the real residue in the table and its sign

(simple pole) is the real pole and its sign shown in the table across from the residue

---

Note Placing a complex conjugate residue over a simple pole creates a non-Hermitian symmetric transfer function. These should not be synthesized. Also, table conversions are not allowed.

Reconstructing Complex Conjugate Pole-Pairs

For complex conjugate pole-pairs, you must associate poles with their residues to reconstruct the pole-residue equation from the synthesis table. A complex-conjugate pole pair and its associated residue is always on the same line in the synthesis table.

For pole-residue tables, the sign of the imaginary part of a complex residue is now significant. Unlike complex pole terms, the imaginary part of a complex residue can have a sign.

The formula for a complex residue over a complex pole is:

$$\frac{(\text{Residue, real part}) + j\,(\text{Residue, imaginary part})}{s - (\text{Pole, real part}) - j(\text{Pole, imaginary part})} \qquad \text{(first conjugate term)}$$

$$+ \frac{(\text{Residue,real part}) - j\,(\text{Residue,imaginary part})}{s - (\text{Pole,real part}) + j(\text{Pole,imaginary part})}$$

where: (real part) is the real number and its sign

(imaginary part) is the imaginary number and its sign; for poles, consider it positive

Associating the sign of the imaginary part of a complex residue with the First Conjugate Term establishes the sign convention for the imaginary part of the complex conjugate pole-pair. Notice that the second conjugate term is formed by conjugating the residue and pole in the First Conjugate Term.

In the case of a simple (real) residue over a complex conjugate pole-pair, the formula used is:

$$\frac{(\text{Residue})}{s - (\text{Pole, real part}) - j(\text{Pole, imaginary part})}$$

$$+ \frac{(\text{Residue})}{s - (\text{Pole,real part}) + j(\text{Pole,imaginary part})}$$

where: (residue) is the real residue in the table and its sign

(real part) is the real number and its sign

(imaginary part) is the imaginary number and its sign; for poles, consider it positive

In this case, the residue has no imaginary part. Thus, there is no sign association between the residue and the complex conjugate pole-pair.

The following tables show how to reconstruct the First Conjugate Term for all possible sign entries in the synthesis table. Since both the real and imaginary parts of the residue can be minus, there are four possible table entries for residues (see table 21-1). Since only the real part of the pole can be minus, there are only two possible table entries for poles (see table 21-2). Table 21-1 shows the numerator constant and table 21-2 shows its denominator for example table entries.

Table 21-1. Reconstructing the First Conjugate Term: Residue

Table Entry	Numerator Constant (residue)
.5 ±j 1.0	.5 + j 1.0
−.5 ±j 1.0	−.5 + j 1.0
−.5 ±j −1.0	−.5 − j 1.0
.5 ±j −1.0	.5 − j 1.0

Table 21-2. Reconstructing the First Conjugate Term: Pole

Table Entry	Denominator (pole)
.6 ±j 2.0	s − .6 − j2.0
−.6 ±j 2.0	s + .6 − j2.0

Building Pole-Residue Equations

As an example, look at the synthesis table in the figure below. Notice that the table contains the following:

- One simple pole and an associated simple residue
- One complex conjugate pole-pair and an associated complex conjugate residue

Using the preceding formulas, the equation for this table is as follows (you can generate the *second conjugate term* by conjugating the *first conjugate term*):

(real part in table) *(imaginary part in table)*

$$H(s) = \frac{(-209.24)}{s-(-250)} + \frac{(104.61) + j(-263.077)}{s-(-500) - j(2000)} + \frac{(104.61) - j(-263.08)}{s-(-500) + j(2000)}$$

$$H(s) = \frac{-209.24}{s+250} + \frac{104.61 - j263.077}{s+500-j2000} + \frac{104.61 + j263.08}{s+500+j2000}$$

(firstconjugateterm)

Task 1c: Entering Data in Polynomial Format

1. Press [**Analys**],

   press [SYNTHESIS]

   press [CONVERT TABLE].

Before you can enter data, you must specify the table format you're going to use, *if you do not already have an existing table in that format.*

2. Press [CONVRT TO POLYNMIAL].

   If the table is already in the polynomial format, skip this step.

   Now press [EDIT TABLE].

This tells the analyzer that you want to enter a synthesis table using the polynomial format.

For polynomial format, the synthesis table is split into two columns: the left side for the numerator polynomial and the right side for the denominator polynomial. Table entries are interpreted as coefficients of the s polynomial.

The first line in the synthesis table corresponds to the numerator and denominator coefficients for s^0, the second line corresponds to s^1, and the nth line corresponds to s^{n-1}.

The highest-order polynomial entry you can make to the synthesis table is a 20th-order polynomial.

3. If you don't need to use the default **1s ^ 0** terms, press [CLEAR TABLE].

   These terms appear when you convert and empty table to the polynomial format.

The default **1s ^ 0** term simply puts a "1" in both the numerator and denominator of the synthesis equation.

4. Using the numeric keypad, enter the first coefficient into the **NUMERATOR** column.

   If you make a mistake, move the highlighted bar over the entry you want to change. Then use the [CHANGE VALUE] softkey to modify your entry.

   To add a denominator, use the knob to move the highlighted bar to the **DENOMINATOR** column.

   You can also use the [ADD VALUE], [DELETE VALUE], and [UNDELETE VALUE] softkeys.

   To clear the table, press [CLEAR TABLE].

5. Press [**Freq**].

   Using the appropriate softkeys, set the start and stop frequencies for the synthesis.

   Once you set these, you don't have to set them for subsequent synthesis calculations.

6. *This step is optional.*

   Press [SYNTHESIS SETUP].

   Using the appropriate softkeys, enter parameters for gain factor, time delay, or frequency scale. You can also toggle the X-axis between linear and logarthmic.

7. You can now run the synthesis.
   To do so, go to Task 2.

The synthesis trace is stored in a dedicated register (data register D8 is the default register, though you can change this). You can run the synthesis without saving the trace to disk. However, the data register is overwritten the next time you create a new synthesis trace.

To save your synthesis table to disk, press [**Save / Recall**], press [SAVE MORE], and press [SAVE SNTH TABLE]. Then using the alpha keys (or an external keyboard), enter an appropriate filename.

To save your synthesis trace to disk, press [**Save / Recall**], press [SAVE DATA], press [SAVE TRACE], and press [INTO FILE]. Then using the alpha keys (or an external keyboard), enter an appropriate filename.

Hints on Entering Polynomial Data

To express your transfer function in polynomial format, use the following formula:

$$H(s) = K \, \frac{(a_1 s^{0} + a_2 s^{1} + a_3 s^{2} + \ldots a_n s^{n-1})}{(b_1 s^{0} + b_2 s^{1} + b_3 s^{2} + \ldots b_m s^{m-1})}$$

where: K is the system Gain

This formula is the basis for interpreting polynomial data in the synthesis table. Polynomial coefficients are in Hertz. During synthesis, the analyzer replaces s with jf (where f is frequency in Hertz).

Polynomial Example

This is an example of polynomial data that you can enter into a synthesis table. It's the same data as in the pole-zero and pole-residue examples, but expressed as polynomial data. To enter this polynomial example, follow the steps in Task 1c. When you are done, your table should look like the one in figure 21-3.

You can expand a pole-zero transfer-function equation to form the polynomial ratio:

$$\text{Pole-zero equation: } H(s) = \frac{1 \times 10^{6}\,(s - 600)}{(s + 250)(s + 500 - j2000)(s + 500 + j2000)}$$

$$\text{Polynomial equation: } H(s) = \frac{1.0 \times 10^{6} s - 600 \times 10^{6}}{s^{3} + 1.25 \times 10^{3} s^{2} + 4.5 \times 10^{6} s + 1.062 \times 10^{9}}$$

Analysis

NUMERATOR 1		SYNTH DENOMINATOR 3	
-600e+6	s^0	1.062e+9	s^0
1e+6	s^1	4.5e+6	s^1
		1.25e+3	s^2
		1	s^3
TIME DELAY = 0 s		GAIN = 1	
FREQUENCY SCALE = 1			

Figure 21-3. Synthesis Table for Polynomial Example

Task 2: Performing the Synthesis

Before completeing this task, make sure you've entered data into the synthesis table, as shown in task 1a, 1b, or 1c.

1. Make sure Trace A is the active trace.

   If it isn't, press [**Active Trace**] once.

   This ensures that the analyzer puts the synthesized frequency response data on Trace A.

2. Make sure you have the proper span selected. To set the span, use the [**Freq**] menu.

   For example, you could set the span to 12.8 kHz with a start frequency at 0 Hz—this is the span we used for this task.

3. Press [START SYNTHESIS].

   If you don't see the [START SYNTHESIS] softkey, press [**Analys**] and then [SYNTHESIS].

   When you start the synthesis, the analyzer plots the synthesized frequency response data on the active trace.

4. Press [**Scale**], then toggle to [AUTOSCALE **ON** OFF].

   It may be necessary to autoscale the trace.

5. Press [**Disp Format**],

   In a moment, you will look at both the magnitude and phase for the synthesized trace.

   press [UPPER / LOWER],

   This displays two traces on the analyzer's screen.

   press [**Active Trace**].

   This makes Trace B the active trace.

6. Press [**Analys**],

   press [SYNTHESIS],

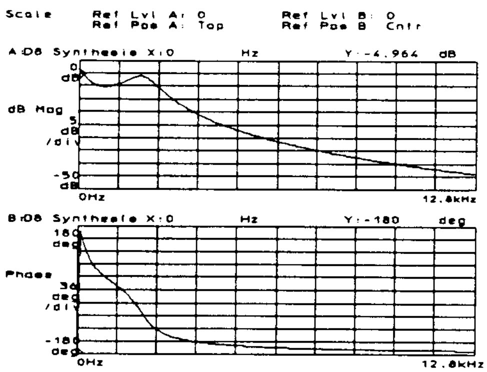
   press [START SYNTHESIS].

   This performs the synthesis and shows the result on Trace B.

7. Press [**Trace Coord**] and

   press [PHASE].

   This changes the trace coordinate of Trace B from dB magnitude to phase.



It may be necessary to autoscale the phase trace. To do this, press [**Scale**] and then toggle to [AUTOSCALE **ON** OFF].

Task 3: Converting Synthesis Table Formats

1. Press [**Analys**],

   press [SYNTHESIS],

   press [CONVERT TABLE].

You can change the data format in a synthesis table to another format with the [CONVERT TABLE] softkey. This softkey brings up a menu with three softkeys—one softkey for each data format.

2. Press [CONVRT TO POLE ZERO] if you want to convert the table to pole-zero format.

   Press [CONVRT TO POLE RESD] if you want to convert the table to pole-residue (partial fraction) format.

   Press [CONVRT TO POLYNMIAL] if you want to convert the table to polynomial format.

Avoid converting tables unnecessarily. Table conversions are not exact because of finite precision in the math operations. It may not always be possible to convert from one representation to another and back, without resulting in slight variations.

The numeric range for tables is full IEEE 64-bit double precision. Note that the display of numbers only handles floats. You can observe full precision by moving the highlighted bar over any term with "infnty" and pressing [ADD VALUE]. Double precision is then given in the numeric entry window.

The message "Not Hermitian" is displayed if coefficients from Pole Residue or Polynomial synthesis-tables are in a table when table format conversion is attempted. The analyzer's curve fitter will not work with a non-Hermitian frequency-response trace (because it is not possible for our pole-zero format to have non-Hermitian table entries).

Task 4: Transferring Synthesis Tables to Curve Fit Tables

1. Press [**Analysis**],

   press [SYNTHESIS],

   press [CONVERT TABLE],

   press [CONVRT TO POLE ZERO],

   press [RETURN],

   press [CURVE FIT],

   press [COPY FROM SYNTHESIS].

These conversion steps aren't necessary if your data is already in the pole-zero format.

To transfer a pole-residue or polynomial synthesis-table to a curve-fit table, you must first convert the data in the synthesis table to pole-zero format.

Task 5: Transferring Curve Fit Tables to Synthesis Tables

1. Press [**Analysis**],

   press [SYNTHESIS],

   press [COPY FROM CURVE FIT].

This completes the procedure.

22

Curve Fit

Introduction

This chapter shows you how to use the analyzer's curve-fit feature. The curve-fit feature lets you find a mathematical model—using up to 20 poles and 20 zeros—to closely approximate the frequency response obtained from a measurement or from a synthesis. After performing a curve fit, you can view the pole-zero results in the curve fit table.

This chapter explains how to use the curve-fit features and steps through some basic tasks common to many curve-fit procedures. If you need information about a specific softkey, see the *HP 35665A Operator's Reference*.

What is Curve Fit?

Curve fitting is a powerful analytical tool that is essentially the opposite of synthesis. With synthesis, you enter a known transfer function model into a table and create a frequency response trace. With curve fitting, you start with a frequency response trace and find a model that corresponds to that trace.

Curve fitting is often done in conjunction the analyzer's math functions. Math functions let you multiply transfer function traces to simulate cascaded systems.

Curve fitting derives a linear system model from either measured or synthesized frequency response data. The pole/zero model is developed by calculating a weighted least-squares fit of the frequency response data to a rational polynomial. You can curve fit using frequency response data obtained with the analyzer's FFT Analysis mode or Swept Sine mode.

Curve fitting with the HP 35665A is done in the s-domain. This means that the curve fitter finds H(s) models, where s is a complex variable with units of Hz. This is of the form:

$$H(s) = K \frac{(a_1 s^0 + a_2 s^1 + a_3 s^2 + \ldots a_n s^{n-1})}{(b_1 s^0 + b_2 s^1 + b_3 s^2 + \ldots b_m s^{m-1})}$$

where: K is the system Gain

Once a curve fit is complete, the numerator polynomial and denominator polynomial are factored to obtain poles and zeros. Only real coefficients are found for the polynomials. Hence, the roots of the polynomials (zeros and poles) appear as real terms and complex conjugate pairs. The results are tabulated in the curve-fit table.

Curve Fit Operating Modes

The curve fitter has two operating modes:

- Fixed Order
- Max Order

In fixed order, the curve-fit routine uses only the specified numerator and denominator order. The analyzer develops a model (with a particular fixed order) with a frequency response that best approximates the complex data in a least-squares sense.

In max order mode, the curve-fit routine uses successively larger system orders until it finds a good model—or until the maximum system order is reached. Both the maximum numerator order (number of zeros) and the maximum denominator order (number of poles) can be specified.

In a moment, you'll learn more about each of these modes.

Fixed Order Mode

Fixed Order Theory

The fundamental component of the curve-fit feature is a non-iterative least-squares complex data fitter. In fixed order, the number of zeros and poles specified under the [CURVE FIT SETUP] menu define the numerator and the denominator order for the complex data fitter. Unlike max-order mode, different numerator and denominator orders are *not* used in an attempt to find an accurate fit. Fixed order allows you to manually experiment with different orders.

Very accurate frequency response matches between the measurement and the curve-fit model are only possible when the order of the curve-fit model is greater than—or equal to—the order of the measured system. Measurement errors from noise and distortion make accurate curve fitting more difficult. Thus, fixed order lets you control the curve-fit model's order when max order has difficulty finding a satisfactory model.

The complex data fitter is deterministic in nature. Therefore, if you repeat a curve fit in fixed order—using the same data and setup parameters—you will obtain the same answer.

Fixed Order Operation

After initial calculations are completed, the complex data fitter uses your system order to find a pole-zero curve-fit model with a frequency response that matches—as closely as possible, in a least-squares sense—the measured frequency response. Since no search for an optimal order is performed, this operation is quite short. The two status lines at the top of the display briefly indicate "fit running" and the current order selection.

When the fit is complete, a synthesis of the curve-fit model appears in Trace B. The status line returns to its idle state.

Max Order Mode

Max Order Theory

Max order mode is the default mode. In max order mode, the max order algorithm operates the complex data fitter (from now on, simply called the *curve fit routine*) with successively larger system orders until it finds a good model, or until the maximum system order is reached. Both the maximum numerator order (number of zeros) and the maximum denominator order (number of poles) can be specified. For example, entering 3 for the numerator upper bound restricts the max order algorithm to finding a model that has a maximum of three zeros.

The max order algorithm starts with a numerator/denominator order of 1,1 and does a fit using the curve-fit routine given this order. The max order algorithm automatically performs a synthesis on the fit model and compares this frequency response to the measured frequency response. If the fit is poor, the orders are incremented to 2,2, and another fit is done. This search upwards in order continues until a good match is found or the upper bounds are reached. The upper bounds do not have to be equal. The max order algorithm holds the order at the first upper bound reached, and lets the other order climb to the higher order if the fits are poor. If both upper bounds are reached before a good fit is found, the max order algorithm returns the fit which came closest to the measured frequency response.

When the curve fitter finds a good fit, the max order algorithm tries to reduce the numerator order and denominator order if it determines that this may create a lower-order model that still provides a close match to the measured frequency response. This order reduction usually results in a numerator order lower than the denominator order.

The max-order algorithm uses the coherence function to determine how large the difference can be between the synthesis of the fit model and the measured frequency response in defining a good fit.

Once the analyzer finds a good fit (or a fit with minimum error is returned, if none of the fits meet the "good" criterion), the numerator and denominator polynomials from the fit are factored into pole/zero form. Then, the max-order algorithm searches for coincident poles and zeros to cancel. Coherence determines how close terms need to be before cancellation can occur. The final results are tabulated in the curve-fit table.

Max Order Operation

When preliminary calculations are finished, the max order algorithm starts with an order 1,1 fit (1 pole and 1 zero). The result is placed in trace B, overwriting the coherence. If the fit is not good, the max order algorithm does an order 2,2 fit and places the result in trace B, overwriting the order 1,1 fit. The current orders are displayed in the upper two status lines as the fit order increases.

When there's a good fit, the max order algorithm checks to see if a reduced order may yield a good fit. This process is called order reduction. Order reduction can lower an order by more than one. If a fit with reduced order results in too large an error, the max order algorithm increments the order and tries another fit. The order is repeatedly incremented (to no higher than where the order was before order reduction began) until there's a good fit. If the incremented order yields a good fit, the max order algorithm again checks to see if order reduction is possible. Occasionally, this causes the curve fitter to repeat two fits. Otherwise, order reduction successively reduces the fit order(s), and then finishes the search process after determining that further order reduction probably won't yield an acceptable fit. Some example sequences for reduction are shown in table 22-1.

Table 22-1. Example sequences for curve-fit order reduction

Example 1	Example 2
(poles,zeros)	(poles,zeros)
5,5	6,6
5,3	6,1
5,1	6,2
complete	6,3
	6,2
	6,3
	complete

When the max order algorithm finishes order reduction, it factors the polynomial-ratio fit into poles and zeros. Then a search is done on the pole and zero terms to determine if any are close enough to each other to be cancelled. If terms can be cancelled, the result display (trace B) is updated with a synthesized transfer function from the remaining poles and zeros. The poles and zeros remaining after cancellation (or all the poles and zeros from factoring if no cancellation occurs) are placed in the curve-fit table along with the gain found.

Note

You can see the synthesis of the intermediate curve-fit results overlaid on the measurement data by selecting the front/back display format (under the [**Disp Format**] menu) before starting a curve fit.

How Max order Defines a Good Fit

To prevent the max order algorithm from always searching to the upper bounds, a variance function is used to establish an error tolerance that gives the max order algorithm some margin when fitting contaminated data.

The max order algorithm uses the number of averages and the coherence function to calculate a variance function on the measurement (you'll learn more about coherence and curve fitting, in "Special Considerations" later in this chapter). The max order algorithm calculates an error-to-signal level, and compares this to a noise-to-signal level. When the error-to-signal level drops below the noise-to-signal level, the fit is defined as good. As a rough approximation, each 0.9 in the coherence function corresponds to –10 dB in the noise-to-signal level. For example, 0.99 corresponds to a noise level at 20 dB below the signal level, and 0.9999 corresponds to a noise level at 40 dB below the signal level. Since the computed variance is used, rather than the coherence function, these figures are merely presented as typical—additional averaging lowers the measurement noise level in the noise-to-signal level calculation.

The curve-fit routine calculates a pseudo coherence when fitting to swept sine measurements (swept sine measurements are available with Option 1D2). Measurement coherence in swept sine is much higher than obtained with FFT-based measurements. A pseudo coherence allows for a more realistic error tolerance in the presence of nonlinearities.

Choosing an Operating Mode

If you know your system order ahead of time, you can specify the number of poles and zeros and perform a fixed-order curve fit. For example, if you want to model your system with two poles, and no zeros, enter 2 for the denominator order (NUMBER POLES), and 0 for the numerator order (NUMBER ZEROS) and perform a fixed-order curve fit. Fixed order will create a 2 pole, 0 zero model that is as close as is possible (in a least-squares sense) to the measured frequency response.

A model for the system you are measuring is ideal, however, and the real system may have resonances or other parameters that correspond to additional poles and zeros. Consequently, you may need to allow for higher orders in your curve fit model. Thus, beginning with an max order curve fit with upper bounds on the system order set several orders (3 or 4) above where you think a reasonable order exists for your system, allows the max order algorithm to automatically find a linear model which comes close to your measured frequency response. You will find that if nonlinearities exist in the measurement, the curve fitter will try to find extra poles and zeros to give a response to closely approximate the effects of nonlinearities on the frequency response.

Typically, you want your model to contain a small number of poles and zeros in order to be practical to work with. Thus, if measurement nonlinearities cause the curve-fit orders to climb unacceptably high, you may want to either lower your system bounds and start the max order curve fit over again, or return to fixed-order curve fitting.

Fixed order is useful when max order returns an unsatisfactory model. For example, the max order fit may increment the system order beyond a point at which you feel the accuracy of the fit has become acceptable. If this happens, you can set the system orders at the point where you saw a good fit and perform a fixed-order fit to find that model. Afterwards, you can try lower numerator orders with fixed order as a way of manually performing order reduction.

Note

Remember, the max order algorithm is essentially an expert system that uses the curve-fit routine as a tool to try and find a good answer. When Max order does not provide an acceptable answer, you can turn to fixed order and use the curve-fit routine directly to experiment with different system orders until you find a reasonable model.

Chapter 22

Curve Fit

The Fit Region

Overview

Curve fitting is always done over a finite range of frequencies. This is referred to as the fit region. The curve fitter (whether in max order or fixed order) attempts to match the measured frequency response in the fit region. The entire frequency response trace displayed on the analyzer is used as the fit region if you select [FULL SPAN] under [FIT REGION]. If you select [USER SPAN] however, the start and stop frequencies under the [FIT REGION] menu determine the fit region.

User span is useful when a fit over the entire frequency response trace does not provide an acceptable fit over some part of the response. For example, the fit may be very good about a sharp resonance, but may begin to diverge from the measurement as the response dips into higher noise regions. If this region is important, you can use user span to restrict the fit region to that portion of the measurement trace where you need a closer match between the fit and the measurement. If this still doesn't provide a good fit, auto weight may not be sufficiently emphasizing the region of importance. In this case, use *user weight* to emphasize the region of importance. Use data edit to set a value (for example, 0.5) to the weighting function over the portion to be emphasized.

Behavior of the Fit Outside the Fit Region

The max order algorithm finds only those poles and zeros needed to provide a good match to the measurement over the fit region. Thus, the frequency response of the curve-fit model outside the fit region tends to follow the behavior of the response near the boundaries of the fit region. Simple poles or complex-conjugate pole pairs (or zeros) can be found outside the fit region if the actual poles and zeros outside the fit region significantly affect the response inside the fit region. Poles and zeros found outside the fit region usually result in a gradual drop-off in the response.

Understanding the Effects of Poles and Zeros Outside the Fit Region

With the following example, we can show the effect that poles and zeros outside the fit region have on the response inside the fit region. We will take the poles and zeros shown in figure 22-1, synthesize them over a span of 25.6 kHz, and fit the data over a smaller region (1.28 kHz to 3.2 kHz).

The poles and zeros shown in figure 22-1 are the same poles and zeros we used in chapter 21 ("Synthesis"), with an additional zero at 65 kHz and a pole at –45 kHz. The additional pole and zero are both outside the fit region but have a small effect on the response inside the fit region. This small effect is not sufficient for the curve fitter to identify the pole at –45 kHz and the zero at 65 kHz—however, the effect is sufficient for the curve fitter to compensate with two new zeros, to replace the pole at –45 kHz and zero at 65 kHz. The curve fit result is shown in figure 22-2. Although the pole-zero models look quite different, figure 22-3 shows the frequency responses to be nearly the same over the fit region.

```
Disk Util   Def Disk: Internal

              POLES  4          SYNTH            ZEROS  2
        -250                              600
        -500  ±j  2e+3                  65e+3
      -45e+3

  TIME DELAY = 0 s                     GAIN = 1e+06
  FREQUENCY SCALE = 1
```

Figure 22-1. Synthesis Model with Extra Pole and Zero

```
Analysis    Strt:  1.28 kHz            Maximum Poles:   20
            Stop:  3.2 kHz             Maximum Zeros:   20

              POLES  3           FIT             ZEROS  3
      -499.96  ±j  1.999e+3       22.879e+3  ±j  26.475e+3
      -249.81                          599.7

  TIME DELAY = 0 s                     GAIN = -0.0011796
  FREQUENCY SCALE = 1
```

Figure 22-2. Small Fit Region on a Synthesis with a Pole and Zero Outside the Fit Region

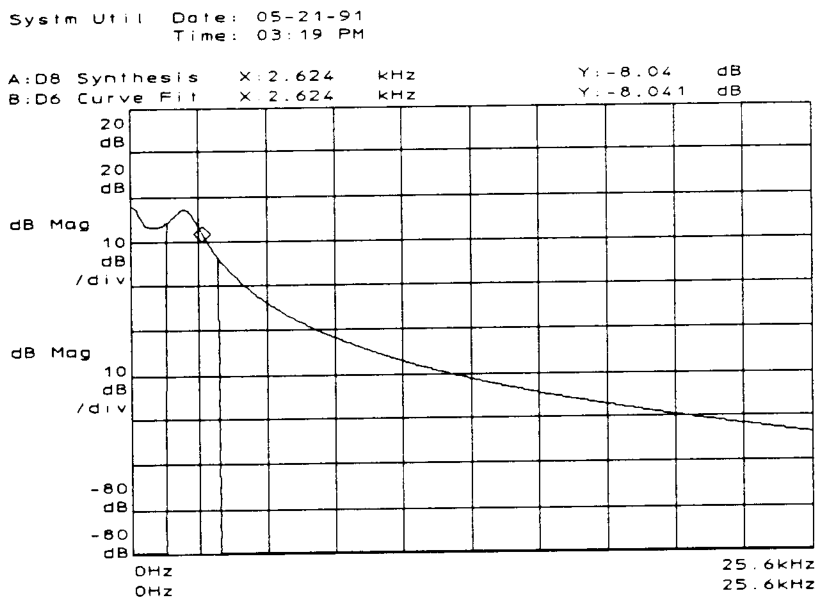


Figure 22-3. Curve Fit with Restricted Fit Region on Over Overlaid with Synthesis

To make this easier to understand, consider the fit we used in this example. As the order of the trial curve fits increased, the curve fit trace was too far away from the synthesized trace to be considered a good fit. When the order reached 3 poles and 3 zeros, the curve fitter found a model whose frequency response inside the fit region was sufficiently close to the synthesis trace to be considered good. At this point, the analyzer used order reduction to try and minimize the numerator order, but without success—in other words, given a limited picture of the complete frequency response, the analyzer found an alternative model that was sufficiently accurate over the fit region.

A comparison of the synthesis model and curve fit model over 51.2 kHz more clearly shows the divergence between the two responses. This is shown in figure 22-4. You can generate this display by changing the span to 51.2 kHz and doing the synthesis with Trace A active. Then copy the curve fit table to the synthesis table, change the synthesis register to D7, and do the synthesis with Trace B active. Then change to the front/back display format. (If the pole at –40 kHz and the zero at 65 kHz aren't present, a curve fit over 1.28 kHz to 3.2 kHz will yield the original model, as shown in figure 22-5.)

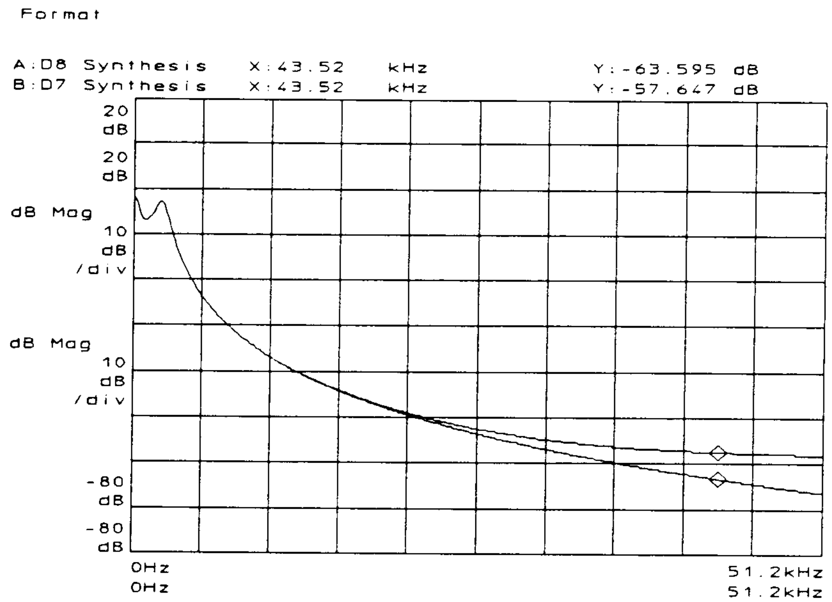


Figure 22-4. Comparison of Curve Fit and Synthesis

Analysis Strt: 1.28 kHz Maximum Poles: 20
Stop: 3.2 kHz Maximum Zeros: 20

POLES 3	FIT ZEROS 1
-500.01 ±j 1.999e+3 -250.01	599.99
TIME DELAY = 0 s FREQUENCY SCALE = 1	GAIN = 1e+06

Figure 22-5. Curve Fit over Smaller Region Yield Original Model

Special Considerations

Why Coherence is Important for Curve Fitting

Coherence is a function of frequency. Each frequency point in the coherence function is a real value between 0.0 and 1.0. A value of 1.0 implies that noise does not interfere with the measurement at that frequency. Averaging allows the measured coherence to converge on the true coherence, which is a measure of the noise in the system, not the measurement. At each frequency point, the coherence and the number of averages are used to compute the measurement variance. The measurement variance is the indication of the noise level in the measurement data. Additional averaging gives a better estimate of the coherence and a smaller measurement variance. Thus, as the coherence at a specific frequency converges to a value farther below 1.0, additional averaging is needed to obtain a small measurement variance at that frequency.

You can improve curve fit results by reducing the measurement variance across the fit region. As an example, if the average value of the coherence across the fit region is 0.9995, only 10 averages may be needed to obtain a good fit. However, if the average value of the coherence drops to 0.95, then 200 averages may be needed for a good fit.

Note

The analyzer uses coherence data (if available) for the curve fit procedure. However, you must have coherence data displayed on Trace B for the analyzer to use it.

Also, keep in mind that coherence data is not available when:

- the analyzer is in Swept Sine mode.
- the data is synthesized.
- the measured data is the result of fewer than four averages.

Note

Overlap processing requires more averaging. However, the increase in measurement speed more than compensates for the additional averaging required for good results. Generally speaking, overlap processing does not affect the curve fitter significantly unless the overlap is more than 90 percent.

To learn more about overlap processing, see the sidebar on "Overlap Processing" in chapter 6.

The Weighting Function

Like coherence, the weighting function is a real-valued function of frequency which varies between 0.0 and 1.0. The weighting function is used to emphasize important areas of the curve during a curve fit and to de-emphasize areas of high variance. The weighting function can be automatically calculated from measured data, or you can create your own weighting function manually.

When you use *auto weight*, the curve-fitter automatically derives a weighting function based on an initial estimate of pole locations and on the variance function. The auto-weighting algorithm tends to emphasize low frequencies and peaks. Thus, as the frequency increases, the emphasis is on resonances (complex poles) rather than simple poles that may change only the rate of amplitude roll-off. This weighting function is placed in the selected weight register during the curve fit.

When you use *user weight*, the curve fitter uses the weighting function stored in the selected weight register. A typical application of user weight is to modify a weighting function created in a previous curve fit using *auto weight.*

When a curve fit using auto weight is finished, you can use data edit to modify the contents of the data register selected as the weight register. Then select user weight and re-run the curve fit. The curve fitter *does not* modify the weight register when user weight is selected.

For synthesized data, the weighting function is not critical to obtaining a good fit, so the auto-weighting algorithm creates a weighting function which has a value of 1.0 at all frequencies.

Under most circumstances the auto-weighting algorithm creates an acceptable weighting function. When the curve fit result doesn't match the measured frequency response in an area that you consider important, it is probably because the auto-weighting algorithm has de-emphasized that portion of the response. In this situation, the best approach to obtaining an acceptable fit is to edit the existing weighting function and try different shapes.

How Coherence Affects the Weighting Function

The max order algorithm uses the number of averages and the coherence function to calculate a variance function on the measurement. Regions of high variance (low coherence, depending upon the number of averages) are not emphasized in the weighting function when you use auto weighting. Also, regions of low variance are not necessarily emphasized: the weighting function only emphasizes regions around peaks and frequencies near dc if the variance is low.

Fitting Measurements With Delay

The curve fitter can have trouble finding a model when pure delay is present in the measurement.

Any pure delay in a system affects the phase of the frequency response without affecting its magnitude. If these delays are known, they should be entered into the curve fit table before starting a curve fit. Pure delays cannot be modeled with a finite-order rational polynomial in the s domain. For more information on the effects of excess phase refer to product note HP 3562A-3, *Curve Fitting in the HP 3562A*.

Fixed Poles and Zeros

When poles or zeros are fixed (using the FIX LINE# key), the curve fitter assumes they are correct and includes them in the next fit. Any pole or zero manually added to the curve-fit table is also tagged as fixed. A fixed term is indicated by "fxd" to the right of the term. The fixed line feature allows the results of one curve fit to be included in the results of another. For example, a curve fit could be performed on a narrow region around a resonance. The resonant pole from this fit would be fixed and another curve fit performed on the entire frequency span. Pole and zero locations found in the first fit and set as fixed are retained and used in the second fit.

Overview of Curve Fit Procedures

Although you don't have to do these steps in the sequence shown here, the following sequence is typical for a curve fit procedure. You can repeat these steps, as needed, to obtain the best fit.

After reading this overview, you can use the sample curve fit tasks—at the end of this chapter—to step through typical curve-fit procedures. To learn more about the material covered in this overview, review the first part of this chapter.

1. *Choose the source of the data.* The analyzer fits the frequency response data displayed in Trace A and places the synthesis of the resulting fit model in Trace B. For best results when curve fitting data measured in the FFT Analysis mode, the curve fitter should use coherence data. Keep in mind that you must have coherence data assigned to Trace B—before starting the curve fit—for the analyzer to use it.

2. *Select max order or fixed order.* If you want a model with a predetermined number of poles and zeros, use the fixed-order mode. If you don't know the orders, use the max order mode. Max order mode uses the specified number of poles and zeros as upper bounds for an automated search for a good fit.

3. *Specify the system order (number of poles and zeros).* In max order mode, the curve fitter experiments with successively larger orders until either a good fit is found or until your specified denominator (poles) or numerator (zeros) order is reached. If your bounds are reached, the best fit found is given. In fixed-order mode, no search is performed. The curve fitter returns a model with the number of poles and zeros you specified.

4. *Specify the fit region (optional).* You can specify a *user span*—a fit region that is smaller than the measurement span. In full span, all measurement data from Trace A is used in the curve fit. Sometimes, you may want to use only a portion of measurement data. For example, you may want to exclude some resonances or regions suspected of heavy distortion. In this case, you can specify a user span—a fit region that is smaller than the measurement span.

5. *Add known poles and zeros to the curve fit table (Optional).* You can edit the curve fit table to add known poles and zeros before you start the curve fit procedure. For example, if your system contains an integrator, add a pole at 0 Hz. This term will be fixed, with an "fxd" indication to the right of the term. Subsequent fits incude fixed terms in the model. The number of pole and zeros set in step 3 must be larger than the order of fixed terms—this provides the degree of freedom in the fit.

6. *Enter any known delays (Optional).* Any pure delay in a system affects the phase of the frequency response without affecting its magnitude. If these delays are known, they should be entered into the curve fit table before starting a curve fit. Pure delays cannot be modeled with a finite-order rational polynomial in the s domain.

7. *Specify the weighting function mode (Optional).* In auto weight, the curve fitter automatically derives a weighting function based on an initial estimate of pole locations and on the variance functions—the curve fitter then places this weighting function in the selected weighting register. The weighting function is used to emphasize important areas of the curve during a curve fit and to de-emphasize areas of high variance. You can use *user weight* to use a weighting function of your own choosing. The selected weighting register is used as the weighting function during the curve fit. Thus, the weighting register is not overwritten during the procedure.

8. *Start the curve fit.* To start the curve fit, simply press [START FIT] under the [**Analys**] menu.

9. *View the curve fit table (and edit if necessary).* To view the curve fit table, toggle to [TABLE **ON** OFF] under the [**Analys**] menu. To edit, press [EDIT TABLE] under the [**Analys**] menu.

Task 1: Basic Curve Fitting

1. Make a frequency response measurement to characterize your device-under-test.

   If you need to review how to do this, see the *HP 35665A Quick Start Guide* or the *HP 35665A Operator's Guide.*

2. Using the [**Meas Data**] menu and [**Active Trace**], put frequency response on Trace A and coherence on Trace B.

   The curve fitter does not use coherence data for the curve fit calculation unless you have coherence assigned to Trace B.

   Press [**Disp Format**] and

   press [UPPER/LOWER].

   It's a good idea to select the upper/lower display format, so you can view the frequency response trace on Trace A and the curve fit trace on Trace B (the analyzer replaces coherence data with the curve fit trace after you start the curve fit).

   Alternatively, you can press [FRONT / BACK] to overlay the results. The vertical scaling for Trace B is automatically set to match Trace A—this makes it easy to compare the two traces.

3. Press [**Analys**],

   press [CURVE FIT],

   press [START FIT].

   Notice how the analyzer replaces the coherence trace with the results from the curve fit.

4. Toggle to [TABLE **ON** OFF].

   You should see a number of poles and zeros that curve fitter selected to characterize the original frequency response measurement. In the next task, you can edit this table, copy it to a synthesis table, and run the synthesis and compare the results to your original frequency response measurement.

Task 2: Editing the Curve Fit Table

Before you do this task, make sure you've already done Task 1.

1. Make sure you are in the [CURVE FIT] menu.

   Then press [EDIT TABLE].

   You should see a number of poles and zeros that curve fitter selected to characterize the original frequency response measurement.

2. Using the appropriate softkeys, change, add, or delete values from the curve fit table.

   Use the knob to move the highlighted bar over the poles and zeros. The changes you make with the editing softkeys occur at the highlighted bar.

3. When you are done editing, press [RETURN] and

   press [RETURN].

   In a moment, you will perform a synthesis on the edited curve fit table to compare the results to your original frequency response measurement.

4. Press [SYNTHESIS] and

   press [COPY FROM CURVE FIT].

   This copies to curve fit table to a synthesis table. You cannot run a synthesis directly from a curve fit table—you must first copy it to a synthesis table.

   Remember, copying to a synthesis table destroys the existing synthesis table (if there was one).

5. If Trace B is not active, press [**Active Trace**] to make Trace B the active trace.

   The analyzer puts the synthesized frequency response on the active trace. To keep the original measurement data on Trace A, we want to put the synthesized data on Trace B.

   Then press [START SYNTHESIS].

   Now you can compare the synthesized results to your original frequency response measurement.

Index

Obtaining a Good Model

The fundamental assumption in curve fitting is that the measured frequency response corresponds to a finite-order rational-polynomial (linear) model. There are several challenges involved in making accurate measurements which fulfill this assumption, and in utilizing measurements inevitably subject to contamination and finite frequency span to obtain reasonable models.

There are three basic sources of data errors in frequency response measurements that prevent any curve-fit algorithm from easily finding a linear model:

- Nonlinearities
- Noise
- Quantization Errors

Nonlinearities

Frequency response measurements may be contaminated by distortion products introduced by system nonlinearities. This contamination can cause the max order algorithm to search for sufficiently high system orders to compensate for these errors in the fit model. For example, although a measured system may contain only 4 poles and 3 zeros, errors may perturb the frequency response enough to look like a system with 10 poles and 10 zeros. In other words, there is no way in which the max order algorithm can differentiate between measurement errors and correct data (fixed order will also do its best to fit measurement errors with the numerator order and denominator order you set).

The type of source you use determines the success of averaging to reduce the impact of nonlinearities on the curve fitting. Broadband stimuli are used for FFT Analysis mode. There are two different types of broadband stimuli used in this analyzer:

- Random noise
- Periodic chirp.

If a truly random source is used (that is, *random noise* or *burst random*), nonlinearities will cause distortion products to randomly appear across the measurement span. Therefore, averaging reduces the effects of nonlinearities on the frequency response measurement, and results in a linear least-squares estimate of the system frequency response.

When using the *random noise* source, you must apply a Hanning window to the time-domain data to reduce leakage in the frequency domain. However, multiplication in the time domain corresponds to convolution in the frequency domain. This means that the frequency response is convolved with the fourier transform of the Hanning window, which produces slight frequency smearing (the main lobe of the window is approximately three bins in width). This frequency smearing can affect measurements on systems with sharp peaks in the frequency response—frequency smearing moves resonant poles a little further away from the j omega axis. Thus, a better measurement results if you use the *burst random* source with *source trigger* and a *uniform window* (burst random is a self-windowing stimulus when source trigger is used). The Fourier transform of a *uniform window* has a narrower main lobe than that of a Hanning window—this provides the best resolution bandwidth possible for a given span.

The other broadband source available is *periodic chirp*. This source type is completely predictable in that it provides exactly the same stimulus from one measurement average to the next. Thus, nonlinearities produce the same distortion products with each average. Consequently, averaging is of no value in reducing the effects of nonlinearities when a periodic-chirp source is used. In addition, the periodic-chirp source makes coherence very high, giving a false impression of a good measurement when nonlinearities may be significant. The periodic-chirp source works best when the system under test is reasonably linear.

One difficulty with using broadband stimuli in some systems is that a signal which is sufficiently strong to provide good signal-to-noise ratios over one portion of the measured span may be too strong for another portion, causing excessive distortion—and possibly rendering the system-under-test inoperable.

The Swept Sine mode (Option 1D2) can provide a solution to this problem. Stimulating the system with a sine wave allows for control of the signal level at a single frequency (rather than broadband noise) over the frequency response measurement. The *autolevel* feature can be used to automatically adjust the source level, as the sweep progresses, to maintain a constant reference level on one of the input channels. Typically, the input channel chosen as the reference is the channel connected to the output of the system under test. With this configuration, the output of the system is held constant rather than the input. This reduces the overall level of distortion while maintaining the signal-to-noise ratio.

The swept-sine algorithm tends to reject distortion products (harmonics) caused by system nonlinearities. However, nonlinearities can also cause errors in the measurement of the fundamental frequency (the frequency of the source sine wave). One symptom of measurement nonlinearities that can be observed using the max order mode of the curve fitter, is a failure of the curve fitter to stop incrementing system orders after the fit appears to be very good. This can occur on FFT based measurements as well.

Another way to check for nonlinearities is to stimulate the system with a fixed frequency sine wave (in FFT Analysis mode; in Swept Sine mode, you can define a math function to compute the FFT of channel 2 time data) and observe the linear spectrum in the response channel. If harmonic distortion is evident in the spectrum, then the system contains some form of nonlinearity.

Noise

Noise will introduce measurement errors. As in the case of distortion products, the curve fitter cannot differentiate measurement errors introduced by noise from correct data. The effects of noise on FFT-based measurements can be minimized by averaging measurements, or for swept sine measurements, by using longer integration times. This assumes that the noise source is uncorrelated to the stimulus (source type) used.

To completely remove errors due to unwanted noise and nonlinearities would theoretically require an infinite number of averages. Since this is not practical, you must choose a compromise. Thus, some error is always present. Averaging reduces the variance of each point in the frequency response measurement as the number of averages increases.

The curve-fit routine uses coherence (coherence is calculated from the tri-spectrum averaging process) and the number of averages to calculate the variance on the measurement. The variance is then used as an estimate of the mean error on the measured transfer function. This information is used to improve the accuracy of curve fitting in the presence of noise. Additionally, the max order algorithm uses the calculated measurement variance to determine how close the curve fit must come to the measured frequency response to be considered a "good" fit. See the sidebar called "How Max Order Defines a Good Fit" earlier in this chapter.

Quantization Errors

Quantization error—better described as *quantization distortion* rather than *quantization noise*—is present whenever an analog signal is digitized. Quanitization distortion can be averaged, as can distortion caused by system nonlinearities. The dithered 13-bit A/D converter in the HP 35665A provides a tightly-specified spurious-free dynamic range of 72 dB.

For curve fitting, this specification is of lesser significance. The curve fit uses the complete frequency response, with very strong emphasis on peaks in the magnitude response. Thus, one or two spurs 72 dB down will have a negligible effect on the fit. With averaging, the typical noise floor may be in excess of 80 dB below the top of your input range.

To minimize quantization errors, remember not to allow a measurement under-range condition when curve fitting in the FFT Analysis mode. The analyzer can automatically range up (if autoranging is on) *but does not range down.* To ensure that the top of your "80 dB range window" is as close as possible to the peak signal strength, press the appropriate autorange softkey under the [Input] menu. *These precautions aren't necessary in Swept Sine mode because in this mode, the analyzer performs optimal input ranging.*

Limitations of a Finite Measurement Span

Curve fitting is performed over a finite portion of the theoretical frequency response. Unexpected poles and zeros may be necessary to account for the poles and zeros in the system beyond the measurement frequency span that have a significant effect inside the measurement span. For example, a typical op-amp has an open-loop response that begins to roll-off between 10 Hz and 100 Hz due to a single pole. Closing the loop pushes this roll-off farther out (for example, to 150 kHz). Thus, an active filter may be designed to have two poles, but is actually affected by another pole farther out, which may be accounted for in a curve fit depending upon the max order curve-fit error tolerance (determined by the measurement variance) in conjunction with the size of the effect.