HP 3563A
Getting Started Guide

Control Systems Analyzer

HEWLETT PACKARD

HP Part Number 03563-90002
Microfiche Part Number 03562-90202

Printed in U.S.A.
Print Date: November 1989

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

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.)

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.

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.

Table of Contents

Chapter 1

Instrument Overview

1

Instrument Overview

This chapter gives an overview of how the HP 3563A gathers data, processes it, and displays it, as well as how the process is controlled by the front-panel keys. You can also refer to chapter 2, "Measurement Overview," in the *HP 3563A Operating Manual.* This overview should be the basis for understanding the HP 3563A.

Some Terms to Know

Hardkeys are all the keys on the front of the analyzer not including the eight keys at the right side of the display screen (called softkeys). Most hardkeys exist only to display menus next to the softkeys. The operation of the HP 3563A is performed using hardkeys grouped in blocks outlined on the analyzers front panel and eight softkeys located down the right side of the display screen. The hardkeys are grouped by function and perform one of three functions:

- Enable an action (such as starting a measurement or turning on a marker)
- Enter data
- Display a softkey menu

Key Sequence is a listing of key presses required to either display a menu referred to in the text or to execute a softkey command. They begin with a hardkey (shown in bold) and have slash marks between key names.

Example: **MEAS DISP**/FILTRD INPUT/LINEAR SPEC 1

Softkey Menus refer to the lists of softkey names shown on the screen down the right side. These change when you pass a key that displays a new menu.

Softkeys are the unlabeled keys aligned in a vertical column at the right side of the display screen. See figure -. They coincide with items in the menus. Softkeys perform several functions:

- 1-of-N selection (N items shown in a left-bracket with current selection bright and underlined)
- Call other menus (has a right-pointing arrow in upper right-hand corner)
- Begin data entry (a data-entry softkey has a large question mark (?) at upper-right of the key name; it "expects" key presses in the numeric keypad; usually terminated by units selection)
- Terminate data entry with selection of units
- Toggle (a 1-of-2 selection with current selection bright and underlined)

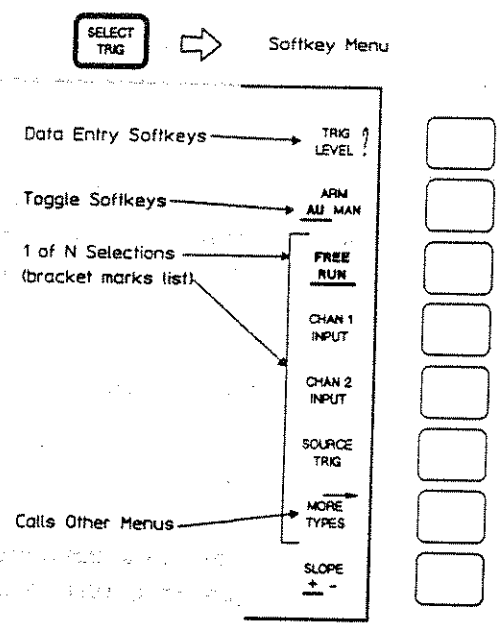


Figure 1-1. Softkey Example: Select Trigger Menu

Front-Panel Tour

The following summaries are an introduction to the basic function of each front-panel key groups or **blocks.** These summaries are presented in the order that the blocks would typically be used to perform a measurement.

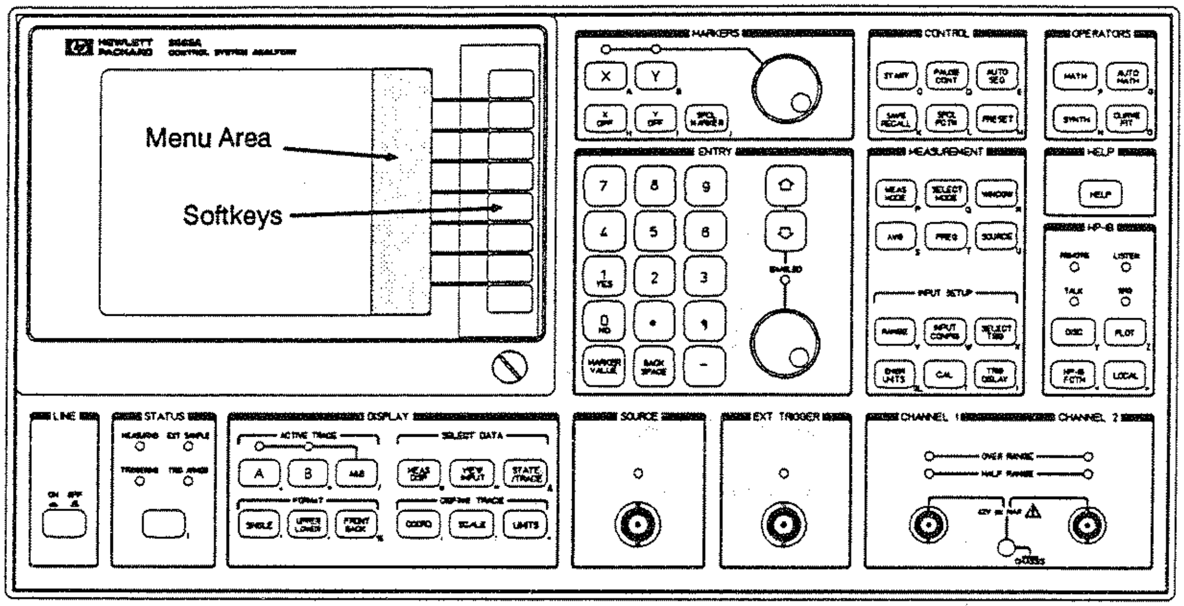


Figure 1-2. HP 3563A Front Panel Introduction

Channel 1 and Channel 2 Input Connectors

⚠ The maximum input signal level allowed on the input connectors is ± 42 V_{pk} relative to chassis ground. Larger voltages could damage the channel input circuitry. The outer conductors of the BNC connectors are **not** connected to chassis ground. This allows the input signal to be floated. They can be individually grounded or floated with selections under the **INPUT CONFIG** hardkey.

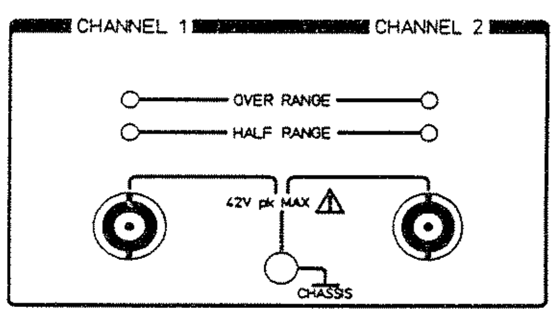


Figure 1-3. Input Connectors

Measurement Block

This group of keys is used as follows:

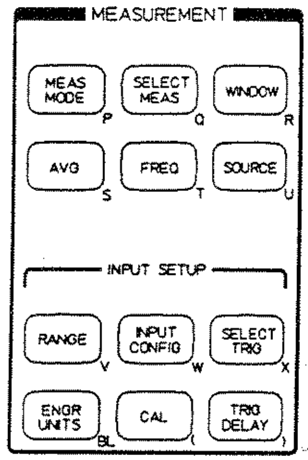


Figure 1-4. Measurement Block

MEAS MODE: selects the measurement mode
SELECT MEAS: selects measurement to be calculated
WINDOW: selects the window used in the FFT analysis
AVG: selects averaging configuration
FREQ: selects the frequency configuration
SOURCE: used to configure the source
RANGE: selects input range configuration
INPUT CONFIG: selects analog /digital input configuration
SELECT TRIG: selects trigger configuration
ENGR UNITS: selects special engineering units
CAL: controls the analyzer's internal calibration
TRIG DELAY: selects the trigger delay

Display Block

A wide choice of display formats and coordinates enhances the analysis of measurements. Depending on the selected measurement, several functions can be displayed. For example, if the selected measurement is FREQ RESP, the measurement display selections include power spectrum and frequency response.

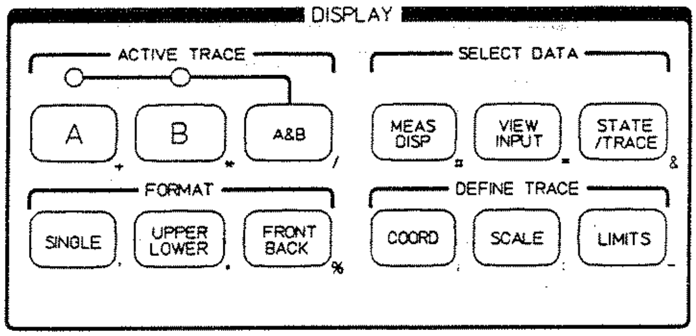


Figure 1-5. Display Block

Active Trace Group: Three hardkeys used to select the active trace. You can have either one or both active. The active trace is used in conjunction with other keys to define/configure the trace data, coordinates, scale, active markers, etc.

Format Group: Three hardkeys used to select the display format. You can have one or both traces displayed. If two are displayed, they can be overlaid with **FRONT BACK** or displayed separately with **UPPER LOWER**.

MEAS DISP: used to select the measurement data to display; depends on selected measurement (configured with **SELECT MEAS**). This hardkey represents roughly half the possible data display selections; the other selections are available under VIEW INPUT. See "The Measurement Process" later in this chapter.

VIEW INPUT: used to display data that occurs before the measurement calculation. This is particularly useful for time-domain data. See "The Measurement Process" later in this chapter.

STATE/TRACE: used to toggle the display between showing the traces and displaying a table containing the state (configuration) of the analyzer. This table may be two "pages" if a channel is digital.

COORD: selects coordinates of measurement data display. Measurement data is complex (in other words, containing both real and imaginary parts). Coordinate selections include magnitude, phase, real, and imaginary, as well as logarithmic or linear and Nichols or Nyquist formats.

SCALE: selects the scaling of the data on the screen.

UNITS: selects the units for both the horizontal and vertical axis. Also adds trace titles.

Markers Block

Markers simplify the analysis of displayed data. Marker functions include single-point and band (delta; Δ) cursor operation. Special markers (such as gain and phase margin, peak search, harmonic, and sideband markers), and slope readouts save time in network and spectrum analysis.

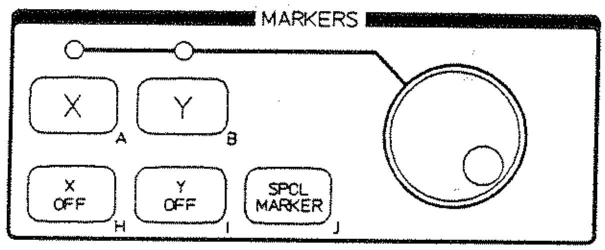


Figure 1-6. Markers Block

X: turns on or activates the horizontal axis marker(s).
X OFF: turns off the horizontal axis marker(s).
Y: turns on or activates the vertical axis marker(s).
Y OFF: turns off the vertical axis marker(s).
SPCL MARKER: selects special marker functions and calculations:

- Harmonic, sideband, slope ,and move-marker-to-peak
- Gain & phase margins, frequency and damping, power, average value, and data editing

Entry Block

You can enter discrete frequencies and levels using the numeric keypad. If the X-axis marker is active, the **MARKER VALUE** hardkey enters the displayed marker frequency value for the active parameter. The up/down arrow keys and the knob are used for fast entry or adjustment of numerical parameters. For example, the knob makes it easy to scroll through the available frequency spans for rapid setup of zoom measurements. Manual selection of input range is simplified with the arrow keys.

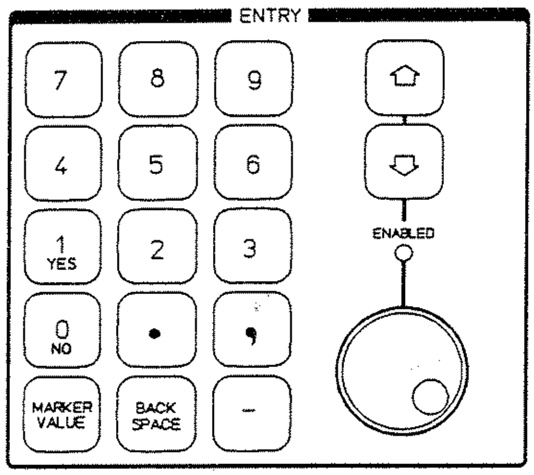


Figure 1-7. Entry Block

Operators Block

These four hardkeys access softkeys that perform advanced analysis of measurement data.

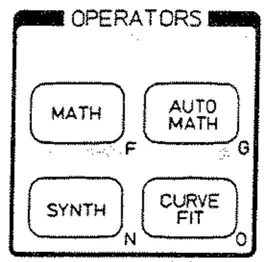


Figure 1-8. Operators Block

MATH: performs waveform math block operations such as algebra, integration, differentiation, forward and inverse Fourier transforms.
AUTO MATH: used to automate math calculations or perform repeated math on measurements as data is collected (See AUTO MATH in the **MEAS DISP** menu).
SYNTH: performs frequency response trace synthesis from data entered in the synthesis table.
CURVE FIT: performs curve fitting of measured or displayed data.

Control Block

This group of keys helps control the analyzer's operation.

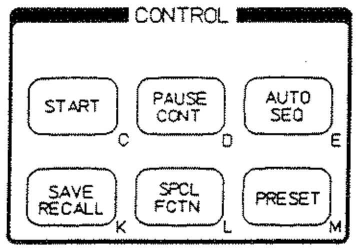


Figure 1-9. Control Block

START: used with **PAUSE/CONT** to start, pause, and continue measurements.
PAUSE/CONT: used with **START** to start, pause, and continue measurements.
AUTO SEQ: used to set up and control auto sequence programs.
SAVE RECALL: used to save and recall analyzer state or data — there are five storage locations for each, numbered 1 through 5. You can also recall the state at last power shutdown.
SPCL FCTNS: accesses a number of the analyzer's miscellaneous features, such as:

- Time and date settings
- Beeper control (on/off)
- Visual help features
- Self-tests and service tests

PRESET: presets the analyzer to the current measurement mode and displays the special preset menu. To perform a complete reset to power- on conditions, press the RESET softkey.

HP-IB Block

These keys allow configuration of the analyzer to provide direct control of external HP-IB plotters and disk drives, for documentation of measurement or analysis results.

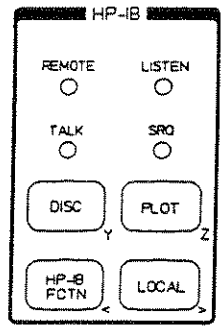


Figure 1-10. HP-IB Block

DISC: selects disk functions
PLOT: selects plotter functions
HP-IB FCTN: used to configure HP-IB (addresses, SRQs, and bus messages)
LOCAL: requests local (front-panel) operation when analyzer is under remote control

Status Block

The operating status of the analyzer is displayed by the LEDs in the Status block. Manually triggered measurements are initiated with the **ARM** key.

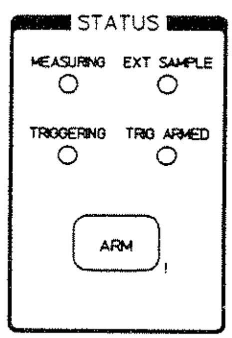


Figure 1-11. Status Block

Help Key

The **HELP** hardkey provides quick, easy to find information shown on the analyzer's display. To get help, press the **HELP** key for general information. Then press the key of interest to get help for that specific key.

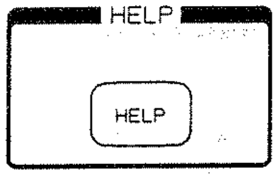


Figure 1-12. Help Block

As an example, press the **HELP** hardkey twice. This displays the help text for the HELP key (see figure 1-13). Note, in the upper right-hand corner of the display, the "Page 1 of 24" callout. This tells you where you are in the help text and how large the help "document" is. To see the next page, press the down-arrow in the Entry Block.

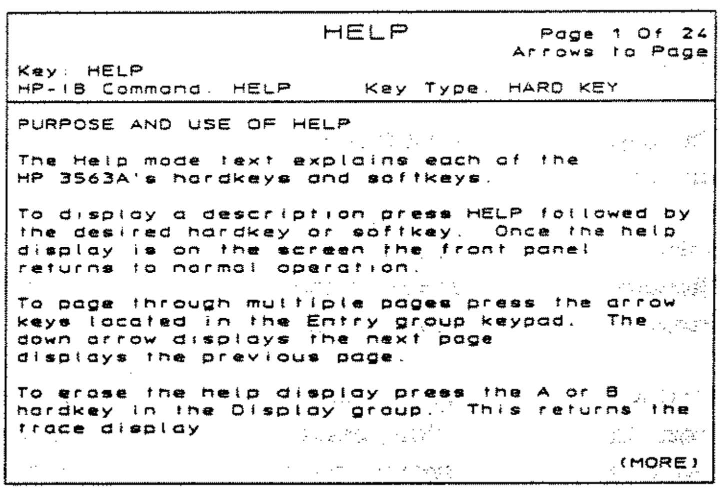


Figure 1-13. Help Text for the HELP Key

A First Measurement

If you haven't used this analyzer before, take a few minutes to make this measurement. This exercise lets you do the following:

- Look at the power spectrum of a 35 kHz sine wave
- Use markers to analyze the signal

Connect a coaxial BNC cable between the analyzer's source output and the Channel 1 input. Press the following keys:

PRESET RESET	The **PRESET** hardkey is the green key in the Control block. This ensures that the measurement process begins from a known state.
MEAS MODE LINEAR RES	Linear resolution is the default measurement mode. This step is included because measurement mode should always be the first thing you set up.
SELECT MEAS FREQ RESP	Frequency response is the default selected measurement. This step is included because selecting a measurement is the second thing you should set up.
MEAS DISP POWER SPEC1	Power spectrum is also the default measurement display. This step is included to show that the displayed measurement is not necessarily the same as the selected measurement. When the selected measurement is frequency response, many measurement displays are possible.
SOURCE SOURCE LEVEL? 1 V SOURCE TYPE FIXED SINE? 35 kHz	This begins the source setup procedure. Select a source level. Any softkey with a question mark is for data entry. Press 1 on the entry block keypad, then the "V" softkey (terminates entry). Select the type of source from a list of several softkeys. The selection of a fixed sine source type requires entry of the frequency. Press **35** in the entry block keypad, then the kHz softkey.
AVG STABLE (MEAN)	Next we turn on averaging. There are several kinds of averaging to choose from. Choose stable. Note that the analyzer takes ten averages and stops. Use **START** to begin another measurement (which is actually a series of averages).
X	Turn on the X marker. It automatically finds the largest value on the trace. The frequency and signal strength values appear in the marker block at the top left-hand corner of the grid. See figure 1-14.
STATE/TRACE	This key alternates the display between the instrument state table and the measurement (trace) display. When one or both of the input channels are digital, the state table is two pages long.

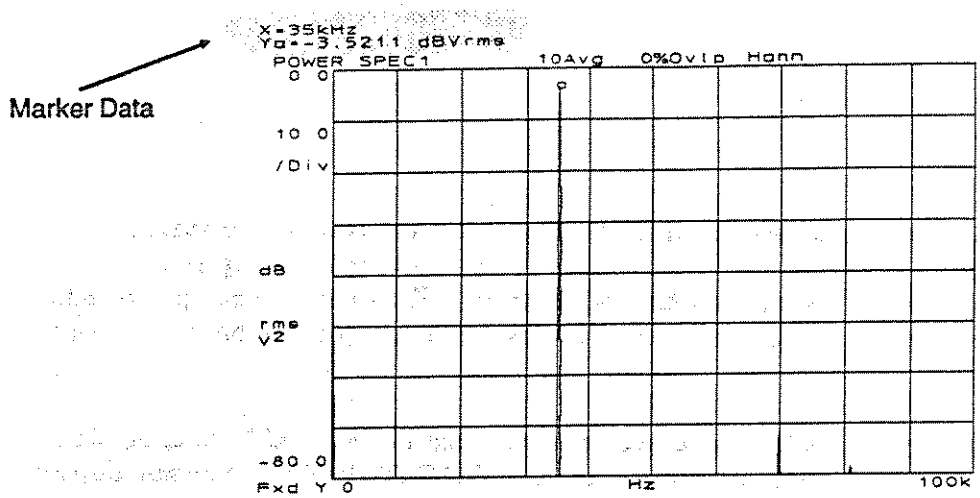


Figure 1-14. Sine Wave Measurement

Linear Resolution

MEASURE	CHAN 1 Freq Resp		CHAN 2 Freq Resp	
WINDOW:	CHAN 1 Hanning		CHAN 2 Hanning	
AVERAGE:	TYPE Stable	# AVGS 10	OVERLAP 0%	TIME AVG Off
FREQ:	CENTER 50 kHz		SPAN 100kHz	BW 187 Hz
	REC LGTH 8.0mS	Δt 3.91µS		
TRIGGER:	TYPE Freerun	LEVEL 0.0 Vpk	SLOPE Pos	PREVIEW Off
INPUT:	RANGE	ENG UNITS	COUPLING	DELAY
CH 1	AutoRngt	1.0 V/EU	DC (Flt)	0.0 S
CH 2	AutoRngt	1.0 V/EU	DC (Flt)	0.0 S
SOURCE:	TYPE Fxd Sin	FREQ 35.0kHz	LEVEL 1.0 Vpk	OFFSET 0.0 Vpk

Figure 1-15. The State Table

The Measurement Process

The HP 3563A measurement process does three things:

- Takes data (makes measurements)
- Processes data
- Displays data

The data may be taken through one or two channels, either of which may be digital or analog, in any combination. The process flow does not always include all the blocks shown in figure 1-16 (for example, swept sine doesn't use the digital filters, windows, or FFT blocks). The displayed data may be processed in any number of ways or not at all, as is shown by the display options on the right side of figure 1-16.

Data displays available "before" digital filtering are selected from the **VIEW INPUT** menu. All data displays available "after" the digital filter are selected from the **MEAS DISP** menu. You may display two of these measurements simultaneously. You can view a simplified version of this process diagram on the analyzer's screen — see "Visual Help."

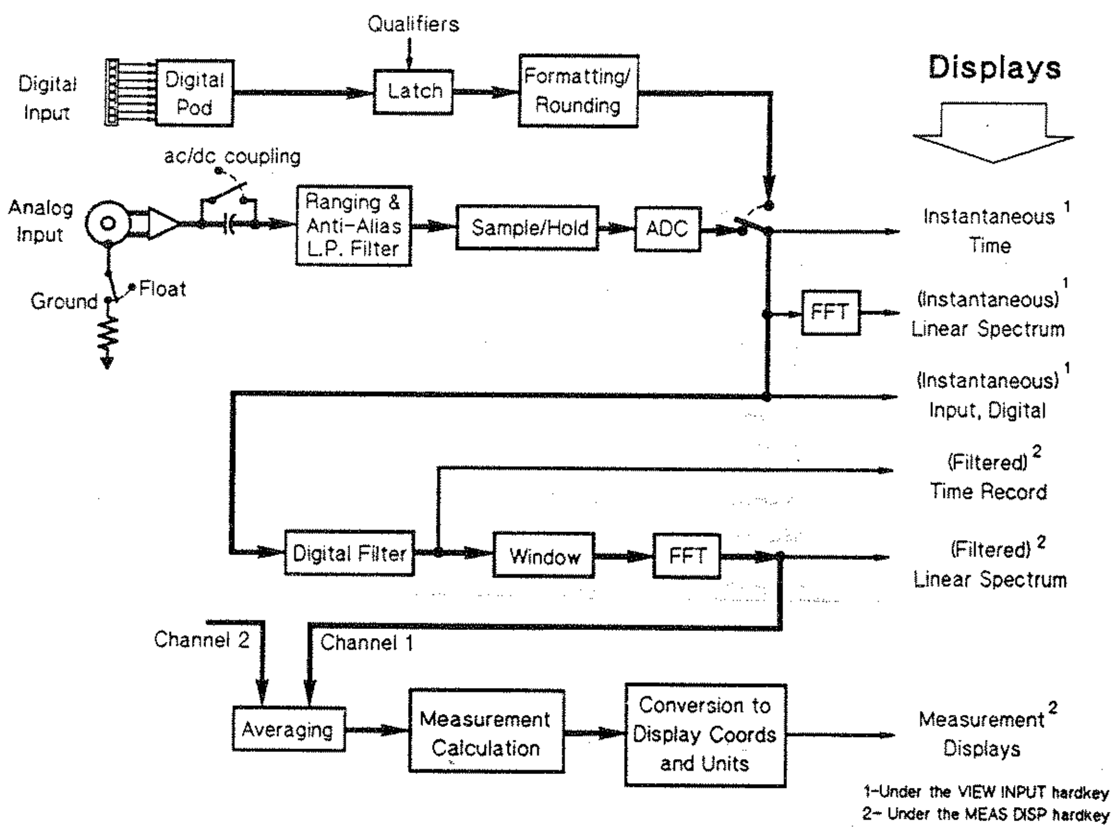


Figure 1-16. Measurement Process Diagram

Basic Configuration

To configure the analyzer, there are five main areas of interest represented by six hardkeys. See figures 1-2, 1-4, and 1-5 for a view of the front panel and location of these keys. Next, each area is discussed in more detail.

Full hardkey name	Abbreviated name
Measurement Mode	MEAS MODE
Measurement Selection	SELECT MEAS
Input Configuration	INPUT CONFIG
Frequency Selections	FREQ
Display	MEAS DISP VIEW INPUT

Measurement Mode

The measurement mode is selected from a menu displayed by pressing the **MEAS MODE** hardkey. The softkeys in this menu are used to select the analyzer's fundamental configuration. The measurement mode options are:

- Linear Resolution
- Log Resolution
- Swept Sine
- Time Capture

Linear Resolution is the measurement technique common to all fast-Fourier transform (FFT) analyzers. Typically, an analog (time domain) signal is sampled until a data buffer (called the *time record*) is filled with a fixed number of time samples. Then the FFT algorithm is performed on the data. This creates a frequency spectrum that may be displayed or used as data for other processing. When the measurement mode is Linear Resolution, the instrument is configured as shown in figure 1-17.

Log Resolution uses linear resolution data to create proportional-bandwidth, logarithmically-spaced measurements. Linear resolution data is combined (rather than redistributed), to produce frequency spectrum or power spectrum data with a true log frequency scale. This technique provides less measurement variance than linear resolution in less time than the swept sine technique. When the measurement mode is Log Resolution, the instrument is configured as shown in figure 1-17.

Swept Sine is a measurement technique based on the single-point Fourier transform frequency response analyzer. This mode performs time-domain integration of the input data to implement narrow-band tracking filters. This technique provides excellent noise rejection, but sacrifices measurement speed. It is typically the favored method for control system analysis. When the measurement mode is Swept Sine, the instrument is configured as shown in figure 1-17.

Time Capture is a measurement technique that performs waveform or time-domain analysis — similar to making measurements with a storage or digitizing oscilloscope. It uses the data stored in the time record (creating this data is one of the first steps of the standard FFT process). While the sampling frequency of FFT analyzers is typically 80 times lower than that of waveform analyzers (256 kHz versus 20 MHz) the dynamic range is usually 20 to 40 dB better than waveform analyzers. Also, FFT analyzers provide filtering to prevent aliasing — this prevents data corruption from frequencies greater than half the sampling frequency. When the measurement mode is Time Capture, the instrument is configured as shown in figure 1-18.

See HP Application Note 243, *The Fundamentals of Signal Analysis*, (in the Appendix) for more information on FFT analysis.

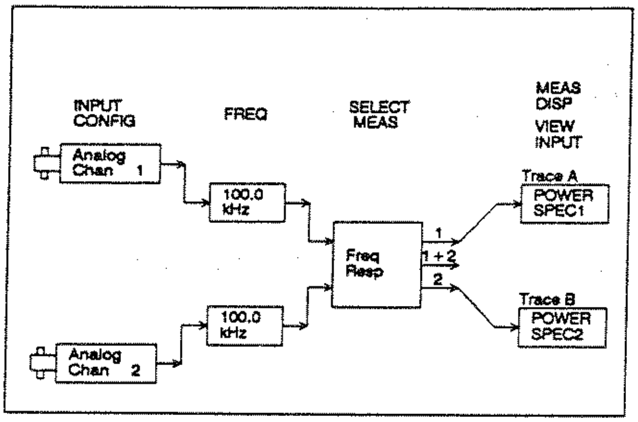


Figure 1-17. A Typical Flow Diagram

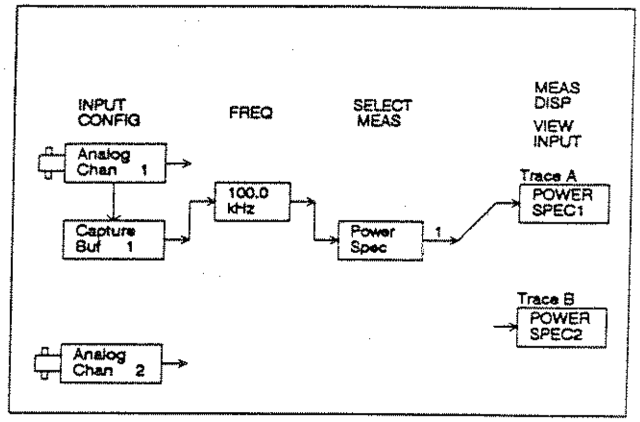


Figure 1-18. The Time Capture Flow Diagram

Measurement Selections

"Select Measurement" is one of the four columns in the flow diagrams of figures 1-17 and 1-18; corresponding softkey selections appear in the **SELECT MEAS** menu. They define how the data is processed in the measurement calculation block of figure 1-16 and the list of measurement selections vary depending on the selected measurement mode. The complete list of selections appears when Linear Resolution measurement mode is selected; they are:

- Frequency Response Function (FRF)
- Power Spectrum
- Auto Correlation
- Cross Correlation
- Histogram

Selecting a measurement mode determines the general configuration of the analyzer's various measurement blocks. It also predefines some of the analyzer's softkey menus because functions necessary or available in one mode may not exist in another. For instance, in the Log Resolution mode, triggering and time averaging are not applicable and windows are not selectable (a predefined window function is used for all measurements), so these menus contain different entries when Log Resolution is active than when Linear Resolution is active.

Input Selections

The Input Configuration is controlled through the **INPUT CONFIG** menu shown in figure 2-1. If both input channels are analog, the selections are very straightforward. When one or both of the input channels are digital, there are many considerations to make and visual display screens to help you make them. Chapter 2 is devoted to a discussion of the digital details of operation.

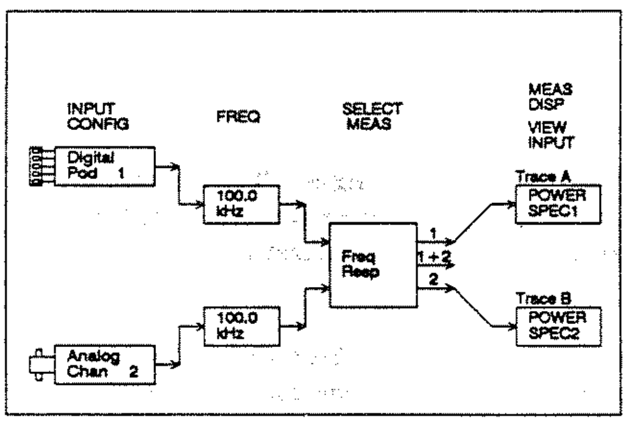


Figure 1-19. Flow Diagram Shows A Digital Input

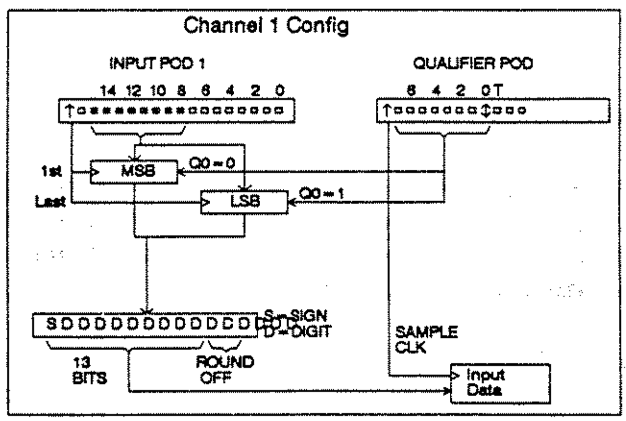


Figure 1-20. Digital Channel Configuration

There are several types of visual help available in the HP 3563A. In addition to illustrations in some of the help text, there are "flow diagrams" and input "channel configuration diagrams" that you can access through the Special Functions hardkey. The key-sequence that displays the visual help menu is **SPCL FCTN**/Visual Help. (The upper-case, bold lettering indicates a hardkey name.)

Flow Diagrams

Flow diagrams show the internal configuration of the analyzer using an illustration shown in figure 1-19. These illustrations are simplified versions of the measurement process diagram shown in figure 1-16.

There are five things to consider first when configuring the analyzer. Thet are:

1. What blocks are used and how data flows through them (in **MEAS MODE** menu)
2. The measurement calculation (in **SELECT MEAS** menu)
3. The filtering selections (in the **FREQ** menu)
4. The display selection (in the **MEAS DISP** and **VIEW INPUT** menus)
5. The input configuration (in the **INPUT CONFIG** menu)

List items 2 through 5 correspond to the four (vertical) areas of the visual help flow diagram shown in figure 1-19. Note that there are two hardkey names at the top of the right-most area of this figure. This indicates that display selections appear in two menus.

Note

The display blocks at the right side of the screen (under MEAS DISP and VIEW INPUT) are not as interactive as the rest of the diagram; when the configuration is changed they do not update until the **START** hardkey is pressed. This preserves measurement data in the display traces as long as possible.

Channel Configuration Diagrams

Channel configuration diagrams help you visualize the digital input configuration. These diagrams are available in two menus; one under the Special Functions hardkey and the other under the Input Configuration hardkey (if an input channel is digital; press the INTERFACE softkey and the CHANNEL CONFIG softkey).

Figure 1-20 is an example configuration that has 16-bit data on an 8-bit bus. To do this we have to monitor Q0 (the least significant qualifier bit) to determine which byte is on the data bus. In this example, the most significant byte is read when Q0 = 0, and the most significant byte is read first (indicated by the First/Last labels).

Other information illustrated by the Channel Diagram shows that the sample clock is provided on the qualifier pod. Other options include channel clocks or the external sample BNC connector on the rear panel. Also, notice that the analyzer word (13 bits) is taken from the most significant bits and that the value of the remaining 3 bits is rounded into the value taken (not truncated). Rounding effectively provides an additional half-bit of resolution.

Displays

You can set up two displays (A and B traces) to display any of the results in the following lists. Their coordinates are selected from the menu displayed under the **COORD** hardkey (to display phase, select **PHASE** in this menu after setting the trace's **MEAS DISP** to be MAG). The general categories of display groups are shown down the right side of figure 1-16.

Specific selections are made from softkey menus under two hardkeys: **MEAS DISP** and **VIEW INPUT**. The group under **MEAS DISP** are displays of data after the data has passed through the digital filter block. Displays listed under the **VIEW INPUT** hardkey are used to display unfiltered data.

Display Type Versus Domain

Display Types	Linear Resolution	Logarithmic Resolution	Swept Sine	Time Capture
Time-Domain Measurements				
Filtered Time Record	X			X
Compressed Time Buffer (1-10 records, Ch 1 or 2)				X
Orbits (Ch 1 versus Ch 2)	X			
Input Time Record (full span, Chs 1 and 2)	X	X	X	X
Auto Correlation (Chs 1 and 2)	X			
Cross Correlation	X			
Impulse Response	X			
Frequency-Domain Measurements				
Input Linear Spectrum (full span, Chs 1 and 2)	X	X	X	X
Filtered Linear Spectrum (Chs 1 and 2)	X			X
Power Spectrum (Chs 1 and 2)	X	X	X	X
Power Spectral Density (PSD, Chs 1 and 2)	X	X		X
Square Root of PSD (Chs 1 and 2)	X	X		X
Energy Spectral Density (ESD, Chs 1 and 2)	X			X
Cross Power Spectrum	X	X	X	
Frequency Response (linear frequency spacing)	X		X	
Frequency Response (log frequency spacing)		X	X	
Coherence Function (with averaging)	X	X	X	
Amplitude-Domain Measurements				
Histogram (Chs 1 and 2)	X			X
Probability Density Function (PDF, Chs 1 and 2)	X			X
Cumulative Density Function (CDF, Chs 1 and 2)	X			X

Chapter 2

Digital Details

Digital Details

This chapter covers the details specific to making connections and selections prior to using the digital capabilities of the HP 3563A. If you haven't yet used this HP 3563A, you should read this chapter before making your first digital measurement.

Input Configuration

You configure the HP 3563A's inputs with the **INPUT CONFIG** hardkey. Figure 2-1 show the softkeys available under this key. Notice that the softkeys differ for analog and digital inputs. If an input is analog, you need to configure only two things: ac or dc coupling, and floating or grounded inputs. If an input is digital, you need to configure many things, such as number format (twos complement or offset binary), data size, data clock, and sample clock. This chapter shows you how to use these softkeys to configure a digital input.

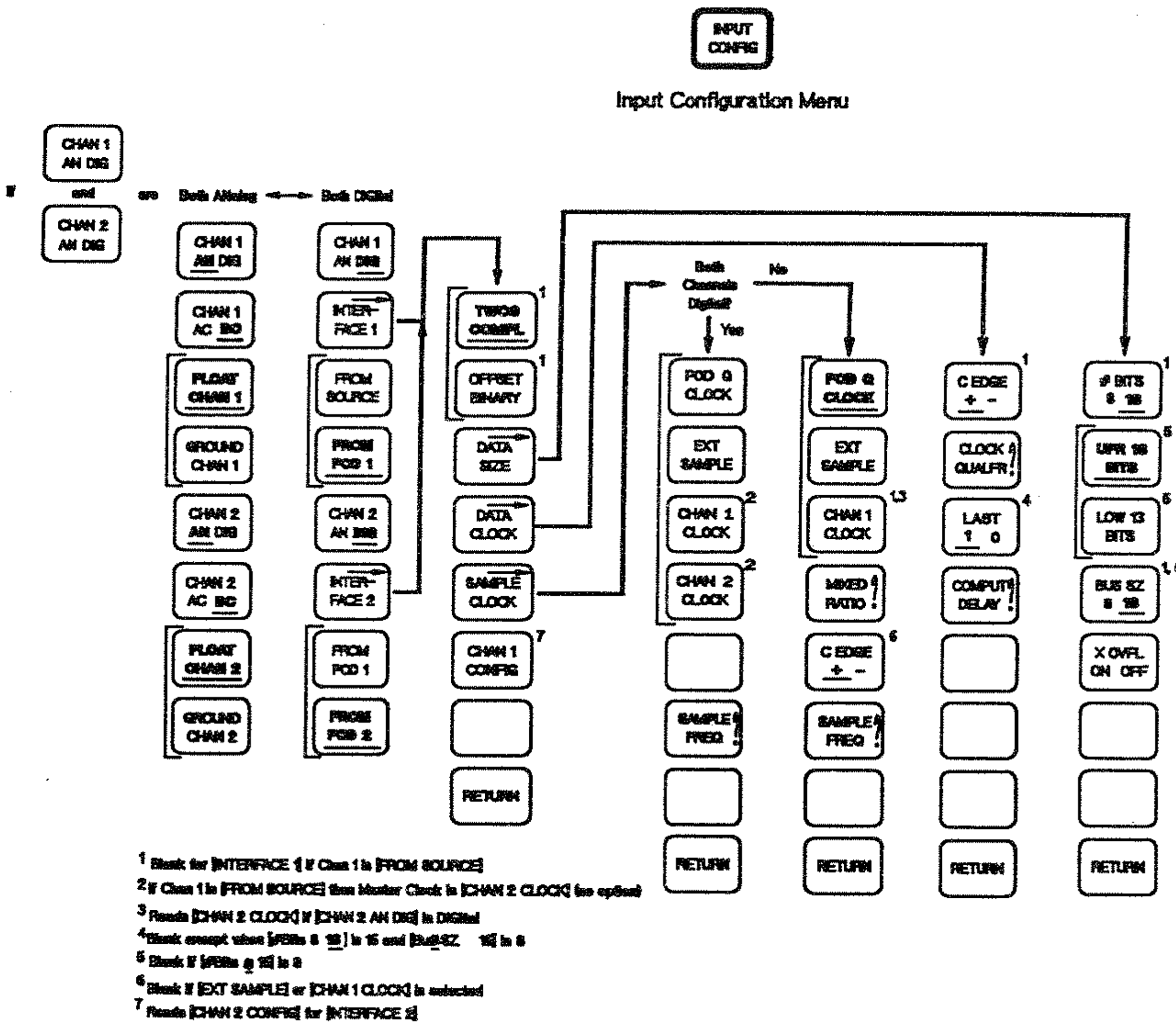


Figure 2-1. INPUT CONFIG Menu Diagram

Configuring an Input Channel for Digital Data

The softkeys in figure 2-1 allow you to configure the HP 3563A's input channel. To configure an input channel for digital data, toggle the channel's AN DIG softkey to DIG. For example, to configure channel 1 for digital data, toggle CHAN 1 AN DIG to DIG; to configure channel 2 for digital data, toggle CHAN 2 AN DIG to DIG. Then use the other softkeys to select the data format, data size, data clocks, and sample clock.

The following sections briefly describe all softkeys used to configure a digital channel (see the *HP 3563A Operating Manual* for complete details).

CHAN 1 AN DIG	Selects analog (default) or digital data for channel 1.
INTERFACE 1	Pulls up the Interface menu for channel 1 (see "The Interface Menu").
FROM SOURCE	Connects Channel 1 to receive digital data directly from the source.
FROM POD 1	Connects Channel 1 to receive digital data directly from Pod 1.
CHAN 2 AN DIG	Selects analog (default) or digital data for channel 2.
INTERFACE 2	Pulls up the Interface menu for channel 2 (see "The Interface Menu").
FROM POD 1	Connects Channel 2 to receive digital data directly from Pod 1.
FROM POD 2	Connects Channel 2 to receive digital data directly from Pod 2.

The Interface Menu

The interface menu allows you to set up a channel's digital interface. It allows you to select the number format (twos-complement or offset-binary), data size, data clock, and sample clock. This menu also allows you to see a visual picture of the current configuration for a channel.

The interface menu is the same for channel 1 and 2. To set up the digital interface for channel 1, press INTERFACE 1; for channel 2, press INTERFACE 2. The softkeys in this menu do the following:

Note The following paragraphs describe the softkeys for INTERFACE 1. These softkeys operate the same for INTERFACE 2, except as noted.

TWOS COMPL — Sets the channel to treat input data as a twos-complement number.

OFFSET BINARY — Sets the channel to treat input data as an offset-binary number.

DATA SIZE — Displays the Data Size menu, allowing you to specify the parameters associated with data and bus size, such as number of bits (8 or 16), size of the Bus (8 or 16), and the significant byte. See "The Data Size Menu."

DATA CLOCK — Displays the Data Clock menu, allowing you to set parameters for an input-pod's CLK signal. The CLK signal and the sample clock are used to clock digital data into the analyzer. See "The Data Clock Menu."

SAMPLE CLOCK — Displays the Sample Clock menu. The data clocks are used to clock data into a data buffer, one sample at a time. The sample clock signal is used to clock digital data into the measurment channel(s). The sample clock is not a unique signal; one of the existing clock signals is chosen to be the sample clock. There are two possible sample clock menus; one if both channels are digital and another if they are not (see figure 2-1). Both menus allow you to choose the signal to use for the sample clock. See "The Sample Clock Menu."

CHAN 1 CONFIG — Displays the channel-configuration diagram for channel 1. For INTERFACE 2, this key reads CHAN 2 CONFIG,and the softkey displays the channel-configuration diagram for channel 2. These softkeys are also available under the **SPCL FCTN** hardkey (see chapter 1).

The Data Size Menu

This menu allows you to specify the parameters associated with data and bus size, such as number of bits (8 or 16), size of the bus (8 or 16), and the significant byte. The following paragraphs give a brief description of these softkeys. See "Data Size" later in this chapter for additional information.

# BITS 8 16	This toggle softkey allows you to choose a data size of either 8 or 16 bits.
UPR 13 BITS	When the input data is 16 bits, the analyzer uses only 13 of the 16 bits. The analyzer uses the upper 13 bits when this softkey is active. The lower 3 bits are rounded.
LOW 13 BITS	When the input data is 16 bits, the analyzer uses only 13 of the 16 bits. The analyzer uses the lower 13 bits when this softkey is active. The upper 3 bits (most significant bits) are ignored. If they are not the same as the sign bit, the message DIGITAL OVER RANGE # (Channel number) appears in the status line.
BUS SZ 8 16	Specifies the size of the data bus being used. Data may be read from a 16-bit bus or an 8-bit bus (see "Data Size" for details).
X OVFL ON OFF	Enables or disables user overflow detection (see "Pod Q Signal Definitions").

The Data Clock Menu

The Data Clock menu allows you to specify the active clock edge and the qualifiers (on Pod Q) necessary to select the data to be associated with the input channel. The following paragraphs contain a brief description of these softkeys. See "Data Clock" later in this chapter for additional information.

C EDGE + −	Selects the active edge for the data clock. The (+) selects the low-to-high transition (the rising edge). The (−) selects the high-to-low transition (the falling edge).
CLOCK QUALFR	Defines the state of the 8 qualifier bits on Pod Q, the qualifier pod, to qualify the clock. Each bit can have the value of 0, 1 or X (don't care). See "Data Clock" and "Pod Q Signal Definitions" for details.
LAST 1 0	Specifies which of two bytes from an 8-bit bus will be the last one read when using 16-bit data. See "Data Size" for additional details.
COMPUT DELAY	Sets a computational delay to correct the time lag associated with the computations in your system from digital filters, microprocessors, 3and the like. See "Computational Delay" for additional details.

The Sample Clock Menu

The sample clock menu allows you to set up your sample clock. There are two possible Sample Clock menus; one if both channels are digital and another if they are not (see figure 2-1). The following paragraphs contain a brief description of these softkeys. See "Sample Clock" later in this chapter for additional information.

POD Q CLOCK	Selects the clock line on Pod Q to be the Sample Clock.
EXT SAMPLE	Selects the external sample, a BNC connector on the back panel, as the sample clock.
CHAN 1 CLOCK	Selects the qualified data clock for Channel 1 to be the sample clock.
CHAN 2 CLOCK	Selects the qualified data clock for Channel 2 to be the sample clock.
C EDGE + –	Selects the active edge for the selected clock. (This choice is available only if the Pod Q clock is the signal selected to be the sample clock.)
SAMPLE FREQ	Allows you to enter the frequency of your external sample clock. This same softkey is also located under the **FREQ** hardkey.
MIXED RATIO	This softkey is used for mixed-domain systems (one channel analog, the other digital). It specifies the ratio between the analog and digital sample rates. See "Mixed-Domain Setup" and "Mixed Ratio" for details.

Rear-Panel Cables

Digital data is transferred into and out of the analyzer with pods connected to the rear panel. See figures 2-2, 2-3, 2-4, and 2-6. There are six cables that plug into the analyzer's rear panel:

- **Pod 1 and Pod 2:** two 16-bit input data pods
- **Pod Q:** 8 qualifier lines, plus a trigger, a clock, and an external overflow signal (all inputs)
- **Source MSB and Source LSB:** two 8-bit source (output) pods
- **Pod X:** a buffered sample clock signal, source clock, and source enable

The input cables must be connected to the device under test (DUT) with the pod tips; the pod tips contain circuitry necessary for proper signal conditioning. The output cable impedance is 50Ω and does not *require* the use of pod tips. Optional accessories that simplify connection to the DUT are:

- HP 10346A 8-channel TTL tristate buffer
- HP 01650-63201 termination adapter: a 40-to-20 pin adapter containing the same signal conditioning circuits as are in the standard pod. This is used to connect the input pods to a 20-pin connector.

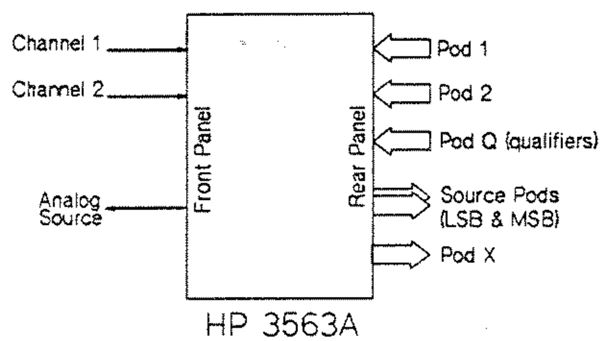


Figure 2-2. HP 3563A Connections

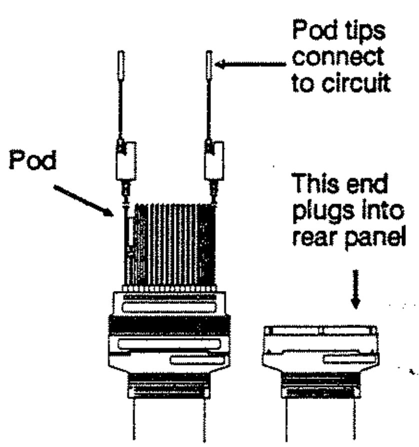


Figure 2-3. Input Cable and Pod

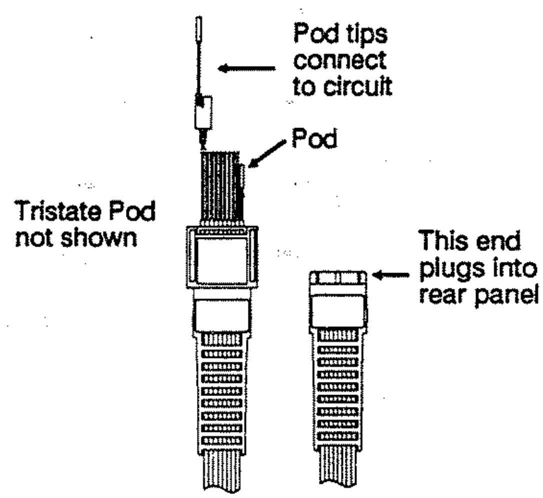


Figure 2-4. Output Cable and Pod

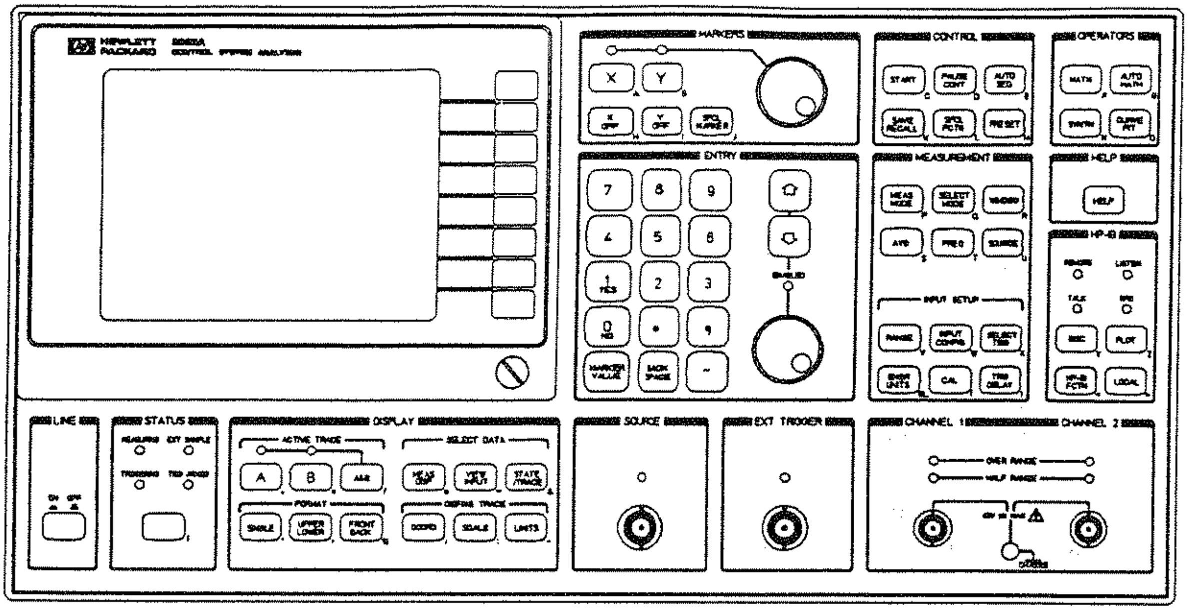


Figure 2-5. Front Panel

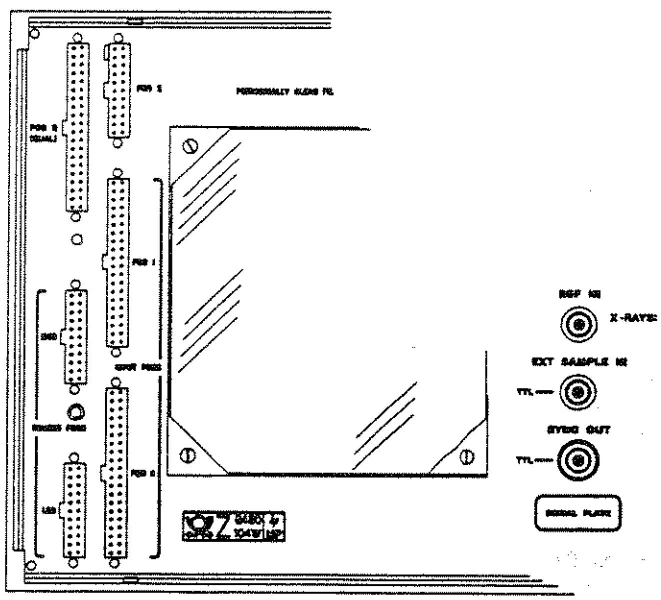


Figure 2-6. Rear-Panel Connectors

Input Pods 1 and 2

Data is brought in through input pods 1 and 2. Each pod has 16 data lines and a clock which must be used to clock in data on that pod. Pod 1 can bring in data for Channel 1 or Channel 2 or both (useful for taking data for both channels from a common bus). Pod 2 can only bring in Channel 2 data. Thirteen of the sixteen data bits are used by the analyzer. You can select either the upper 13 bits of a 16-bit word (in which case the lower 3 are rounded) or the lower 13 (in which case the upper 3 are truncated).[1] You can transfer eight-bit data or sixteen-bit data on an eight-bit bus. The data format can be either two's-complement or offset binary. See figures 2-9 and 2-11 for information on data timing versus the sample and data clocks.

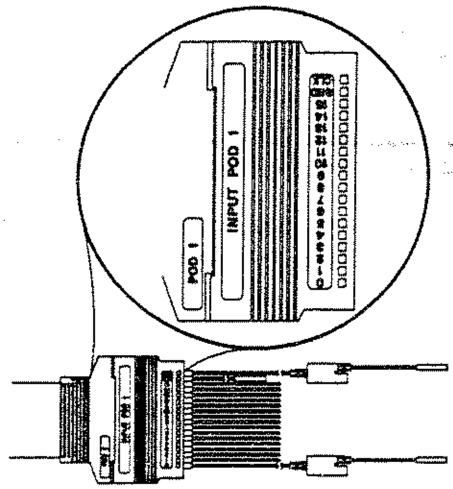


Figure 2-7. Input Pod

Number Format

Twos-complement: sddddddd dddddddd (MSB LSB) where
s = sign bit (0 = positive, 1 = negative) d = digits

Offset binary: sddddddd dddddddd (MSB LSB) where
s = sign bit (1 = positive, 0 = negative) d = digits

Offset-binary number formats are most commonly used with DAC's and ADC's.

Signed Integer Number	Two's-Complement		Offset Binary	
	(hex)	(octal)	(hex)	(octal)
32767	7FFF	077777	FFFF	177777
1	0001	000001	8001	100001
0	0000	000000	8000	100000
−1	FFFF	177777	7FFF	077777
−32768	8000	100000	0000	000000

1 In swept sine measurement mode, the analyzer can be configured to automatically switch between upper and lower 13 bits to provide full 16-bit performance.

Data Size

The number of data bits doesn't always match the number of data lines on the bus. Use the following configuration table to best set up the analyzer. When the (input) bus size is 8 bits and the data size is 16 bits, the analyzer multiplexes the incoming data and reconstructs the data internally. In this case, Q0 is *used to indicate most significant byte* (MSB) by entering a clock qualifier word in the data clock menu as the address (1 or 0) of the MSB. It is assumed, then, that when that state appears on the Q0 line, the data is the MSB. LAST 1/0 (softkey in the same menu) is *the state of Q0 which marks the last byte of the two-byte transfer* when that channel is providing the sample clock. Otherwise, the last two bytes clocked in before the sample clocks will be taken as the data word. These two settings (values of Q0 and Last) allow the two bytes to be read in either order as well as allowing either state of Q0 to indicate MSB. See figure 2-8.

Bus Size	Data Size (# Bits)	
	8	16
8	Connect the 8 data lines to the 8 upper-most pod lines.	1. Connect the 8 data lines to the 8 upper-most pod lines. 2. Connect Q0 on the qualifier pod to a signal to indicate which byte should be clocked in first. 3. Go to Data Clock/Clock Qualfr and set the value of Q0 (1 or 0; qualifier definition appears at the bottom of the display). 4. If that channel is providing the sample clock, use the LAST 1/0 key to select which byte is clocked in last. It may help to turn on the visual display of the digital interface. 5. Select the upper or lower 13 bits.
16	Connect the 8 data lines to the 8 upper-most pod lines	Connect all 16 lines to the pod; then select either the upper or lower 13 bits to be processed.

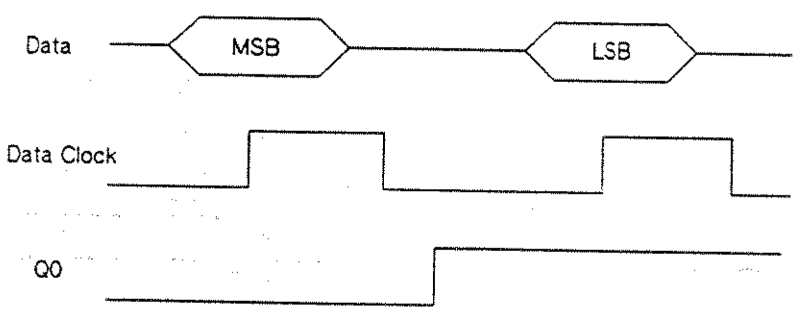


Figure 2-8. Example: Move 16-bit Data on an 8-bit Bus

Data Clock

Often, Channel 1 and Channel 2 data in a sampled data system is not available at the same time. For this reason, separate clocks on input pods 1 and 2 (the CLK signal; see figure 2-7) are used to transfer data into the analyzer. Each pod has a clock which must be used to clock in data from that pod. This clock has the same function as the clock on a logic analyzer. Data is transferred into the analyzer when a clock transition occurs while the qualifiers are in the defined state (transition direction is configurable). This clock is referred to as either a *data clock* or a *channel clock*. If the signal selected for the sample clock is not present, the message "Check External Clock" appears in the lower right corner of the display. Refer to figure 2-11 for timing relationships between the sample and data clocks.

Sample Clock

The analyzer expects sampled data points for each channel to be synchronized to one clock — the *sample clock*. It is not a special input signal. Rather, it is chosen from among the various clock signal inputs. The sample clock can be either Pod Q Clock, External Sample, Channel 1 CLK, or Channel 2 CLK. The frequency range of the sample clock is 0.001 Hz to 256 kHz.

Sample clock is the main system clock — all other timing (and phase) relationships are referenced to it. There can be only one sample clock selected for the analyzer. In the case where there are two clocks in a system (implying that data occurs at different times within a sample period), the sample clock is usually set to be the signal that last clocked in input data. However, a data clock can occur up to 250 ns after the active edge of the sample clock and still have its data point associated with the sample clock edge (see figure 2-9). If the data clock occurs after t_2 but before t_1, the data on that pod may be associated with either sample edge (the relationship is not defined and the measurement data is probably corrupt).

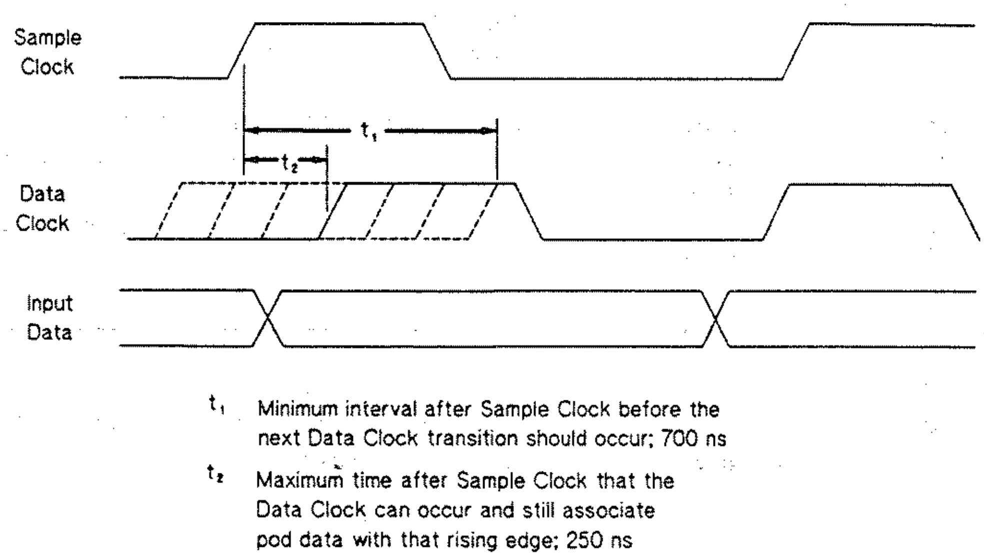


Figure 2-9. Timing Between Sample and Data Clocks

Computational Delay

Computational delay is used to correct for physical delays in sampled data systems. (For example, to accommodate the computation time of a digital filter and delays introduced in the measurement process, such as clock skew).

If you know the delay time between the channels, you can enter this value as the computational delay and an appropriate phase ramp is applied to the selected channel data. When it is applied to a channel that is part of an FRF measurement, a positive value of computational delay added to channel 2 results in a negative-going phase ramp. A positive value of computational delay added to channel 1 results in a positive going phase ramp in the FRF. (FRF = Ch2 / Ch1.)

Note A computational delay can be specified for each digital channel. This doesn't usually make sense but may be used for special cases, like correcting the timing of both channels relative to a clock that isn't related to either channel.

Example: Digital data is clocked into latches with the pod data clocks. Internally, the two digital channels receive data simultaneously on the sample clock's active transition. If there is a known delay between the channels that you need to preserve, you may enter that value as the computational delay. This value is used to adjust the phase calculations for frequency domain measurement data.

Restrictions

1. When bringing in data for both channels on Pod 1, you must wait at least 700 ns after clocking in a data sample on one channel before clocking in data for the second channel. Bringing in data this way requires use of qualifiers for the clock. Also, the same clock edge must be used for both channels on Pod 1.
2. There is a special case for which data may be lost for sample frequencies above 248 kHz. This may occur when both channels are digital and one channel, whose channel clock is being used as the sample clock, is transferring 16-bit data on an 8-bit bus and data is clocked into the other channel within 700 ns of the sample (and channel) clock. When this occurs the error message "MISSING SAMPLE" is displayed.

Note While all pods and channels meet published specifications, Pod 2 can handle a somewhat higher data clock repetition rate and lower data clock pulse width that Pod 1. So if you need to take data on one channel from a faster data clock than the HP 3563A is rated (10 MHz), use Pod 2.

Maximum sample rate is still 256 kHz, of course.

Grounding

Ground (black wire) is a connection to all of the interlaced ground wires on a cable. You should connect this ground to the digital ground of the device under test. Note also that each pod tip has a provision for a ground connection, if the black wire connection is inadequate (due to noise or ground loops). Usually, it isn't necessary to connect both black wire ground and the individual pod tip grounds. Either one should be sufficient.

Mixed-Domain Setup

Mixed-domain refers to the configuration where one channel is analog data and the other is digital data. This requires some special considerations:

- *Always* use View Input to check that you have the expected signals before taking a measurement
- If either channel is digital:
  - You must supply a sample clock (F_s) (see "Sample Clock")
  - The analog channel is clocked by the sample clock (see "Mixed Ratio")
- The digital channel is clocked at or before the analog channel (see "Mixed Ratio")
- There is no anti-aliasing below 100 kHz for the analog channel

Mixed Ratio

If the measurement is mixed domain, mixed ratio can show the effects of a sampled system on the analog portion of the circuit. A sampled system has its frequency response replicated at multiples of the sample rate. These replications (or images) may or may not affect the analog portion of the design. Mixed ratio allows you to measure the system at a sample rate comparable with the analog system, and also simultaneously measure a digital system at a lower rate. This requires using the faster clock as the analog clock, and the (slower) digital clock as a data clock. You enter the digital clock rate (F_{sd}) as the sample frequency and a number between 1 and 512 as the mixed ratio value. (For more information, see "Mixed Ratio" at the end of chapter 3 — particularly as the Mixed Domain Example at the end of this chapter.)

For example, the digital part of a control system runs at 1 kHz. To measure the effects of the digital system over an 8 kHz bandwidth, enter 1 kHz as the sample frequency and a ratio of 8. A 1 kHz clock would be connected as a channel clock and an 8 kHz clock connected to Pod Q clock.

Qualifier Pod Q

Qualifiers

In some applications, data isn't valid at every transition of a given clock signal. For these applications, you can use the eight qualifier lines on Pod Q (Q0-Q7) to clock data into the analyzer. The analyzer monitors the state of these lines and compares them to states you entered at the front panel. When a clock edge occurs and the pattern matches the state of the qualifier lines that you entered, data is clocked into the analyzer. This allows the use of Pod 1 to bring in data for both channels; each channel is given a unique qualifier pattern (similar to an address). This method may also be used to enable source data out of the analyzer when using a tri-state pod.

Qualifiers are not required, but provide flexibility for systems that transfer both input and output data on a single bus (like a microprocessor data bus). For example, a digital filter reads data from a memory-mapped I/O port on one bus cycle, makes a computation, and outputs data to a different port on another cycle. Qualifier bits connected to address lines, used in conjunction with the input-pod data clocks (either Pod 1 or Pod 2) can be used to acquire filter input data on one channel, and output filter data on the other. The clock signals commonly used are address strobe signals, read or write signals, or a CPU clock. To understand how qualifiers can be used to enable source data, refer to the discussion on Pod X. For setup and hold-time information, see figure 2-11.

Other Pod Q Signals

TRIG (digital trigger) is another form of external trigger, but one where the front-panel signal "External Trigger" except that the front panel signal is qualified by slope and level. TRIG is a TTL signal that triggers on the rising edge.

Q-CLK is a clock input that is most commonly used as the sample clock in mixed analog/digital configurations.

OVF is a user-enabled overflow/overload indicator for the system under test. A high TTL level indicates an overload condition. If overload rejection is on (X OVFL is ON; see figure 2-1)while averaging, that data is rejected when an overload condition occurs.

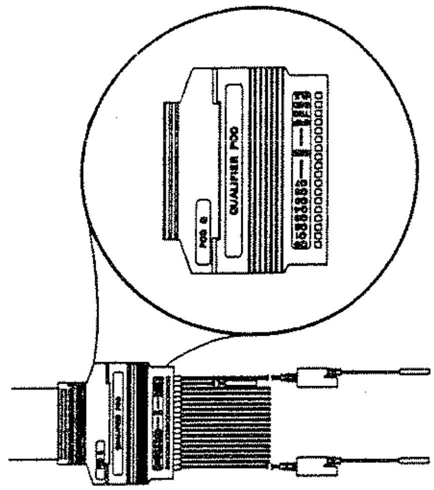
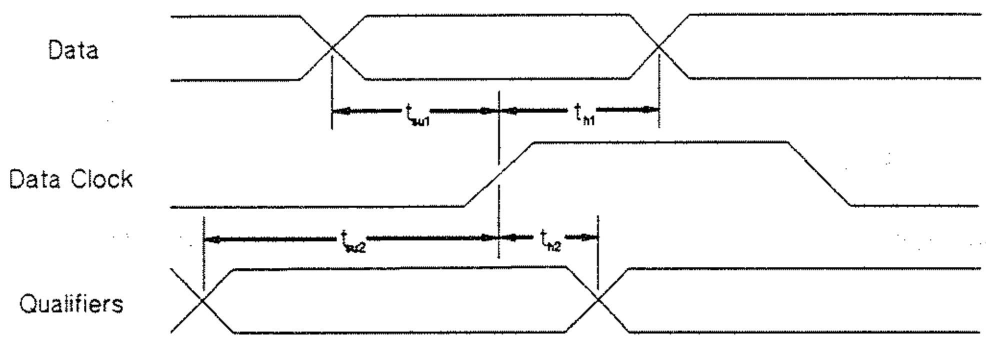


Figure 2-10. Pod Q Signal Lines



t_{su1} Data set-up time relative to data clock; 20 ns

t_{h1} Data hold time relative to data clock; 5 ns

t_{su2} Qualifier set-up time relative to data clock; 60 ns

t_{h2} Qualifier hold time relative to data clock; 5 ns

Figure 2-11. Setup and Hold Times for Digital Inputs

Source Pods MSB/LSB

Sixteen bits of source data are available on two 8-bit pods — source MSB and source LSB. The upper 8 bits appear on the source MSB pod; the lower 8 bits (where bit 0 is the least significant bit) appear on the source LSB pod. These are TTL-level signals with 50 Ω output impedance.

A new source value appears with every occurrence of the sample clock. Source data becomes valid approximately 150 ns after the transition of the sample clock (+ transition of Ext. Sample, or either transition of the Channel Clock or the Pod Q-Clk). A signal on Pod X (SRC-CLK) indicates valid source data. See figure 2-14 and the SRC-CLK description under Pod X.

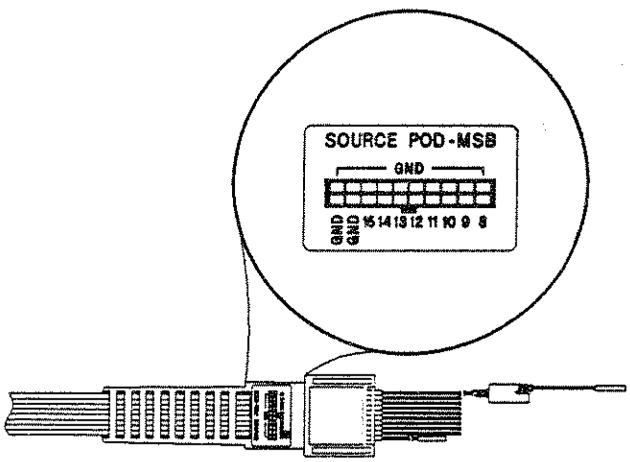


Figure 2-12. Source Pod MSB

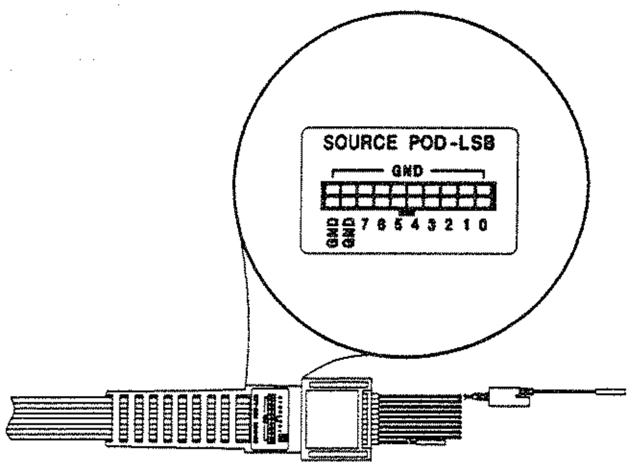
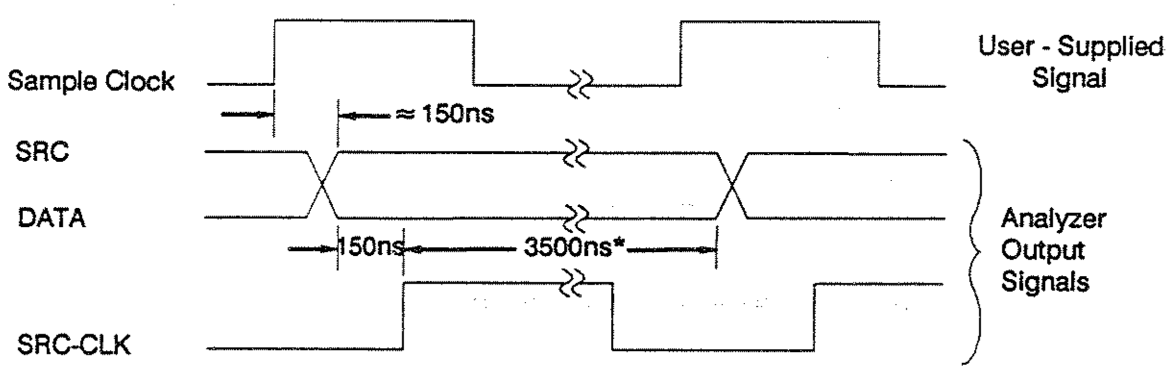


Figure 2-13. Source Pod LSB



* 3500ns is the minimum at a 256 kHz sample rate.
Interval increases with decreasing sample rate.

Figure 2-14. Source Timing Diagram

Pod X

Pod X provides a buffered sample clock signal and two special source signals, as follows:

SRC-CLK (source clock) is an analyzer output that indicates (on its rising edge) when source data becomes valid. The source clock signal becomes active (high) approximately 150 ns after the source data is valid, this occurs 100 to 200 ns after the sample clock. The source clock signal transition occurs once per sample period. See the figure 2-14 for setup and hold time information.

SRC-EN If you are using a tri-state bus, source data can be enabled onto the bus with the tri-state pod accessory (HP 10346A). This pod has built-in tri-state buffers. It requires you to provide +5 volts, ground, and an active-low enable signal. If the system being tested does not have a convenient enable signal, SRC-EN may be generated with qualifiers. SRC-EN should be connected to the tri-state pod enable. Qualifier signals should be connected to Pod Q and configured with the Source/Data Interface softkeys. When the qualified condition is satisfied, SRC-EN goes low and turns on the tri-state pod signals.

SMP-OUT is a buffered version of the Sample Clock signal.

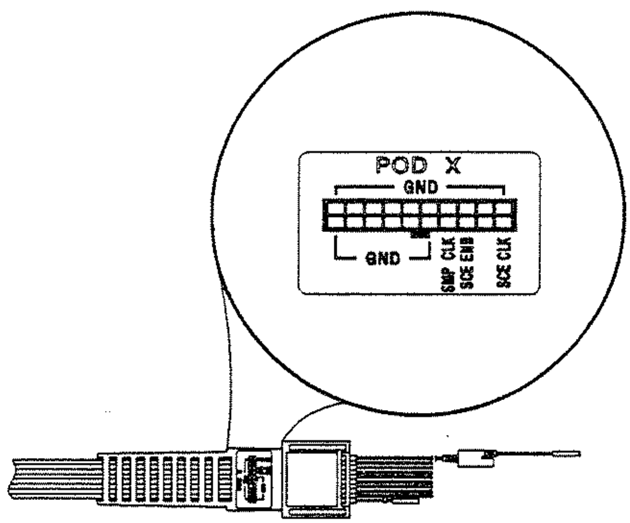


Figure 2-15. Pod X Signal Lines

An All-Digital Example

This section shows you how to interface the HP 3563A to a common signal-processing system. Then, it shows you how to verify correct operation of the signal-processing system's digital filter algorithm. The block diagram for this example appears in figure 2-16.

System Operation

The circuit in figure 2-16 is a common signal-processing system, in this case a Texas Instruments TMS320 Digital Signal Processor. The ADC performs analog-to-digital conversions at a rate F_S. When it finishes a conversion the ADC clocks data into an input port (holding register) and signals the TMS320 that data is ready by asserting ADCRDY*. (Signal names ending with an asterisk denote signals which are active-low.)

The TMS320 reads input data from Input Port 6 and performs a digital filter calculation. When the calculation is complete, the TMS320 writes the result to Output Port 2 by asserting OUTP2*. Output Port 2 is another holding register. Its output is connected to a DAC that converts the digital data back to an analog signal. After the write to Output Port 2, the TMS320 writes status information to Output Port 3.

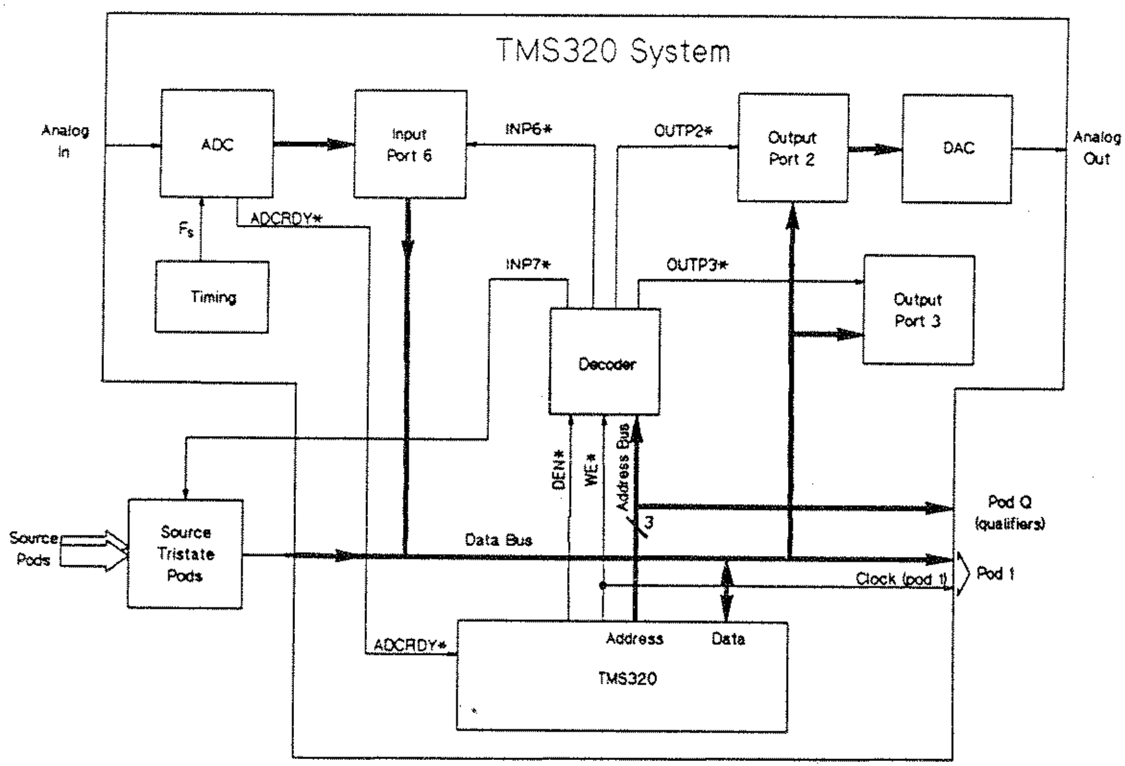


Figure 2-16. Example Digital Filter Block Diagram

The TMS320 I/O structure is designed to map address lines A0-A2 into eight input and output ports. A decoder monitors these address lines and asserts control signals (INP6*, INP7*, or OUTP2*) based on the port address plus DEN* or WE*. DEN* is asserted by the TMS320 during input operations; WE* is asserted during output.

Signal Definitions

Signal Name	Description
A0, A1, A2	Address lines: lowest three TMS320 address lines
ADCRDY*	ADC Ready, active low: signals the TMS320 that a conversion is complete
DEN*	Data Enable, active low: control signal generated by the TMS320 for data input
INP6*	Input port 6, active low: used to enable input data from ADC
INP7*	Input port 7, active low: used to enable input data from tri-state source pods
OUTP2*	Output port 2, active low: used to clock output data into the DAC
OUTP3*	Output port 3, active low: used to clock output data into a status register
F_s	Sample clock: generated by timing circuit
WE*	Write enable, active low: control signal generated by the TMS320 for data output

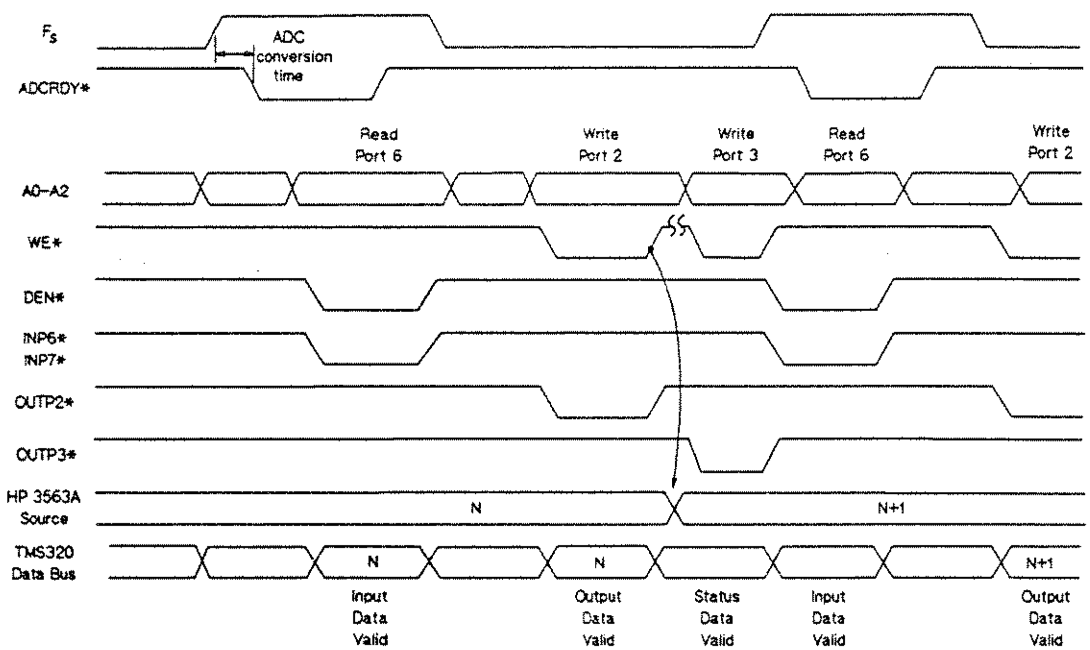


Figure 2-17. TMS320 Timing Diagram

Digital Filter Frequency Response Function

For this exercise, we want to verify correct operation of the digital filter algorithm. We can verify the digital filter algorithm by measuring the frequency response of the digital filter. To do this, we will use the HP 3563A's digital source and one digital channel. This type of measurement is called digital-in/digital-out measurement.

Connecting the Digital Source

First, we must determine where to connect our digital source. From our block diagram, we know that the TMS320 reads data from Input Port 6, performs its digital filter calculation, and then writes the result to Output Port 2. To measure the frequency response of the digital filter — in this case the TMS320 — we need to inject our signal into the TMS320 and see the result of its digital filter calculation. In other words, we need to connect our digital source to the TMS320 data bus.

We must modify our block diagram before we can connect our digital source to the TMS320 data bus. Looking at our block diagram, we see that Input Port 6 — the register that holds ADC data — is tri-state. Furthermore, we see that the TMS320 enables this register after an ADC conversion by asserting INP6*. We need to disable Input Port 6 to avoid bus contention. We can do this in one of two ways:

- Hard-wire the tri-state enable on the ADC register (Input Port 6) inactive, and connect INP6* to the tri-state enable on the source pod accessory.
- Rewrite the TMS320 program to read from a port other than port 6; for example, port 7. Then connect this new signal, INP7*, to the tri-state enable on the source pod accessory.

We will use the second method. In this case, the TMS320 is signaled at the end of an ADC conversion. But instead of reading from Input Port 6, it reads from Input Port 7 (our digital source). This requires a simple one-line change in the TMS320 program. Since the enable time and propagation delay for the tri-state pod is approximately the same as that of the ADC register (18 ns and 12 ns, respectively), we satisfy all timing requirements. Lastly, since the TMS320 data bus is a tri-state bus, we must use the tri-state pod accessory for our digital source (HP 10346A).

Connecting the Digital Channels

Now we must determine where to connect our digital channels. This is easy. For a frequency response measurement, we need to connect one channel to the input of the device-under-test (DUT) and the other channel to its output. For frequency response measurements, the HP 3563A expects the input of the DUT on channel 1 and the output on channel 2. So we must connect channel 1 to the input of the DUT. Since the digital source is connected to the input of our DUT, we can connect channel 1 to the digital source. To do this, we can take advantage of the HP 3563A's internal connection of the digital source to Channel 1 (see "Configuring the Analyzer"). In this case, we don't need to provide a clock for Channel 1 — it is also provided internally.

Now we must connect channel 2 to the output of the TMS320. We can use either Pod 1 or Pod 2 — in this case we'll use Pod 1. Pod 1 can be connected directly to the 16-bit TMS320 data bus. We must also connect a data clock to Pod 1. Looking at the TMS320 timing diagram, we see that output data is valid when the TMS320 writes to port 2. Thus, we could use OUTP2* as the data clock. Or, we could use WE* as the data clock. The TMS320 asserts WE* when it writes data. Note that WE* is asserted twice during one sample period — first to Output Port 2, then to Output Port 3. If we use WE* as the data clock, we must use qualifiers (on Pod Q) to select the correct WE* assertion.

For this example, we will use WE* as the data clock to illustrate the use of qualifiers. We'll connect qualifier lines Q0, Q1, and Q2 directly to address lines A0, A1, and A2. Later, we'll configure the analyzer to accept data when the data clock goes from low-to-high and the state of the qualifier lines is 010 (Port 2).

Selecting a Sample Clock

Now that we've connected our digital source and digital channels, we need to select a signal for the sample clock. As a reminder, a sample clock must be used for all digital measurements. It is important to remember the difference between the data clock and the sample clock. The data clock transfers data into the analyzer; the sample clock is the main system clock, to which all timing (and phase) relationships are referenced. In other words, the data clock is referenced to the sample clock (see "Sample Clock" earlier in this chapter). In the HP 3563A, the digital source outputs a new data-point with each active edge of the sample clock.

We need to look at the TMS320 timing diagram to determine the signal to use for our sample clock. For this example, we decided to use the same clock for both the data clock and the sample clock. This simplifies the connections to our circuit. Remember, for our data clock we selected the rising edge of WE* qualified with address lines A0, A1, and A2 (same time as when OUTP2* goes high). Since our data clock is qualified, our sample clock is also qualified. We can see in the timing diagram the effect of using this same signal for both the data clock and the sample clock. Notice that when both the address lines equal 010 (Port 2) and WE* goes from low-to-high, the following occurs:

- The TMS320 output data is valid
- The digital source changes state (the line labeled "HP 3563A Source" changes state)

In other words, we transfer data into the analyzer via Pod 1 and, at the same time, output new source data via the source pods. Since the source data is fed into a tris-state pod, the source data won't be read by the TMS320 until the TMS320 enables the tri-state pod.

By further examining the timing diagram, we can also see that when the qualified data clock goes high, source point N is associated with response point N. Thus the phase of our FRF measurement will not be affected.

Note

Always examine timing relationships before making a measurement to understand any factors that could affect phase. Timing skew between channels can cause phase artifacts. For example, if the output data for point N – 1 occurred when source value N was present, then a phase ramp associated with this 1-sample-point delay would be introduced in the FRF.

Configuring the Analyzer

Now that we've made all the necessary connections to our circuit, we need to configure the analyzer. We'll press the following keys to do this:

INPUT CONFIG	
CH1 AN DIG	Selects digital data as input for channel 1.
FROM SOURCE	Makes an internal connection between the source and Channel 1.
INTERFACE 1	
CHAN 1 CONFIG	Displays the Channel 1 configuration diagram shown in figure 2-18.
CH2 AN DIG	Selects digital data as input for channel 2.
FROM POD 1	Selects Pod 1 as the source of the digital data for Channel 2.
INTERFACE 2	Displays a visual picture of the Channel 2 configuration. This picture will
CHAN 2 CONFIG	update as we continue to setup the analyzer (figure 2-19 shows final setup)
SAMPLE CLOCK	Notice that the default sample clock is Chan 2 Clock — this is what we want.
SAMPLE FREQ?	Sets the sample frequency to that of our system.
15.625 kHz	Note that the positive (+) edge is default —3 this is what we want.
RETURN	
DATA CLOCK	Selects 010 for qualifier bits Q2, Q1, and Q0 (respectively).
CLOCK QUALFR	Press the DON'T CARE softkey to enter the x's.
xxxx x010	
SOURCE	Note that the source automatically changed to digital.
SOURCE LEVEL	
30 mV	We'll select a source amplitude of 30 mV.
SOURCE TYPE	
BURST CHIRP	Burst chirp works best with this filter.

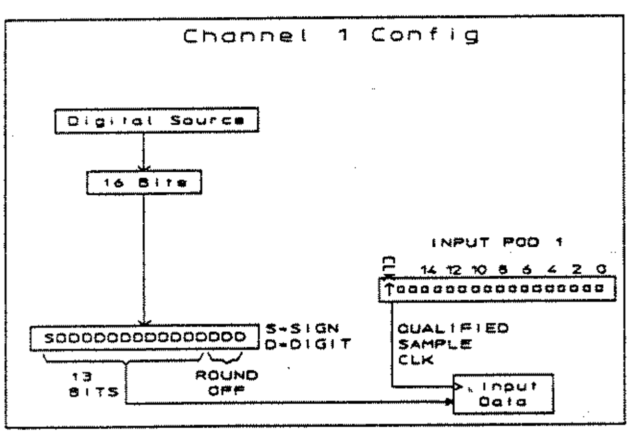


Figure 2-18. Digital Channel 1 Diagram

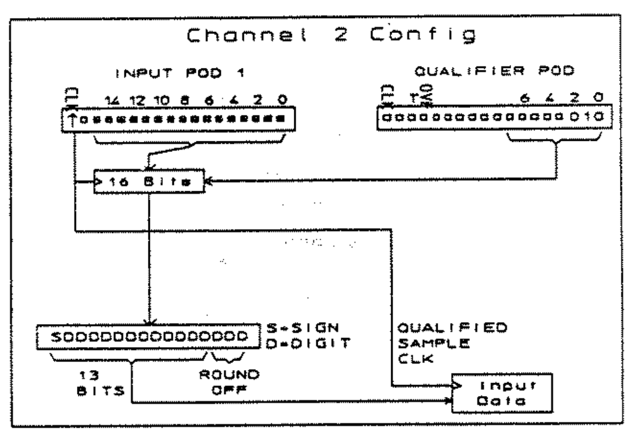


Figure 2-19. Digital Channel 2 Diagram

WINDOW UNIFORM	Burst chirp is self-windowing; no window needed.
LINES 1024	Allows viewing up to $F_s/2$.
SELECT TRIG SOURCE TRIG	Acquires new time record when the source outputs its data.
STATE/TRACE	Displays the analyzer state.
STATE/TRACE	Displays digital setup state as shown in figure 2-20.
STATE/TRACE	Puts the trace back on the display.
MEAS DISP FREQ RESP	Selects the frequency-response measurement display.
SCALE Y FIXED SCALE 40, – 80 dB	Sets the maximum and minimum y-axis scale values to 40 dB and – 80 dB (numeric values, including the minus and comma, are entered via the numeric entry pad; units (dB) are entered via softkeys.)

We now see the frequency response of the digital filter, as shown in figure 2-21. The y-markers indicate a dynamic range of about 80 dB. The digital filter was designed to have a dynamic range of about 100 dB. Our measurement is limited because we can process only 13 bits of data on both channels. To improve our measurement, we can switch to Swept Sine mode and take advantage of the built-in range switching of the digital channels. The analyzer automatically changes to the lower 13 bits when it detects lower level signals. This feature is available only in Swept Sine mode.

```
                    Digital Setup                     Page 2

INPUT:      CHAN 1                      CHAN 2
            Digital (Pod 5)             Digital (Pod 1)

DATA:       FORMAT      # BITS          BUS SZ      ALIGNMENT
  CH 1      Twos Cmp    16              16          Upr 13
  CH 2      Twos Cmp    16              16          Upr 13

CLOCK:      EDGE        QUALIFIER                   DELAY
  CH 1      Pos         XXXXXXXX                    0.0 S
  CH 2      Pos         XXXXX010                    0.0 S

SAMPLE:                 FREQ
  CH 2                  15.6kHz
-------------------------------------------------------------
OUTPUT:     Digital

SOURCE:     TYPE             BURST      LEVEL
            Brst Chirp         70%      29.8mVpk

DATA:       FORMAT      QUALIFIER       RANGE
            Twos Cmp    XXXXXXXX        5.12 Vpk
```

Figure 2-20. State Table, page 2

To set up the analyzer for swept sine mode, we need to press the following keys:

MEAS MODE

SWEPT SINE	Selects the Swept Sine measurement mode.
LINEAR SWEEP	Selects linear sweep (instead of log sweep) for the swept sine measurement.

RANGE

AUTO 1 16 BIT	Selects digital autorange for Channel 1.
AUTO 2 16 BIT	Selects digital autorange for Channel 2.

START	This measurement takes several minutes. See the results in figure 2-22.

Notice that in the completed swept-sine measurement the null region contains some noise. We could improve the measurement in this region by increasing the gain of our source. To do this, we could use the Swept Sine mode's autogain feature to automatically vary the source level to maintain a constant amplitude.

Conclusion

For systems that require digital frequency response measurements with approximately 100 dB dynamic range, Swept Sine mode with digital autoranging is the best solution. Any measurement mode will work for systems that require digital measurements with 80 to 85 dB dynamic range.

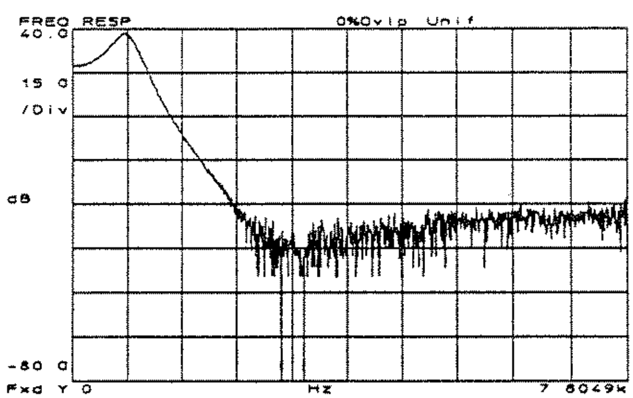


Figure 2-21. 13-Bit Measurement

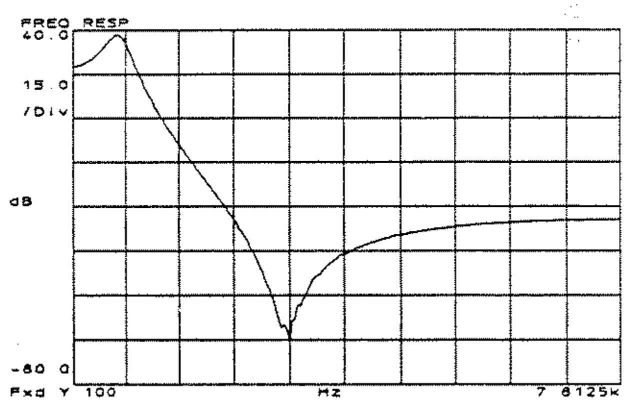


Figure 2-22. 16-bit Measurement

Mixed-Domain Example

The control system shown in figure 2-23 has both analog and digital components. The sampled data portion of the system (ADC, Digital Filter, and DAC) runs at a 31.25 kHz sample rate (F_{sd} is the digital sample frequency). Energy at frequencies above $F_{sd}/2$ causes aliasing in this loop and can potentially affect stability.

Sampling causes spectral images to appear at multiples of the sampling rate (see figure 2-24). These higher frequency signals, if not filtered, will feed back and cause aliasing. An anti-alias filter (low-pass filter) could be inserted before the ADC, but it is desirable to minimize cost in the design.

In general, the plant in a control loop has a low-pass filter characteristic. Also, the DAC has a zero order hold characteristic that functions as a low-pass filter. It's possible that, together, the DAC and plant could provide enough attenuation to eliminate the need for an additional filter. Our objective is to characterize the DAC-plant combination to see if it adequately filters the image frequencies produced by sampling.

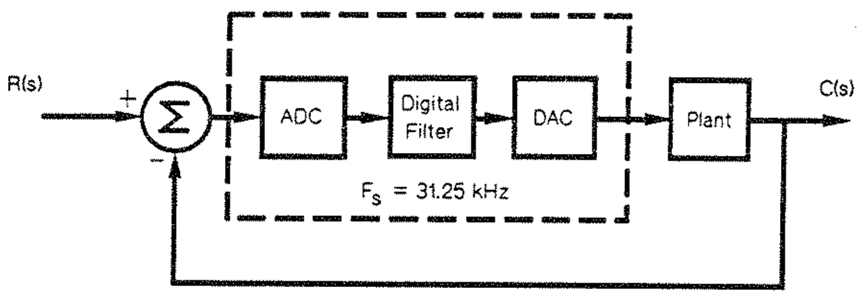


Figure 2-23. Sampled Data System Block Diagram

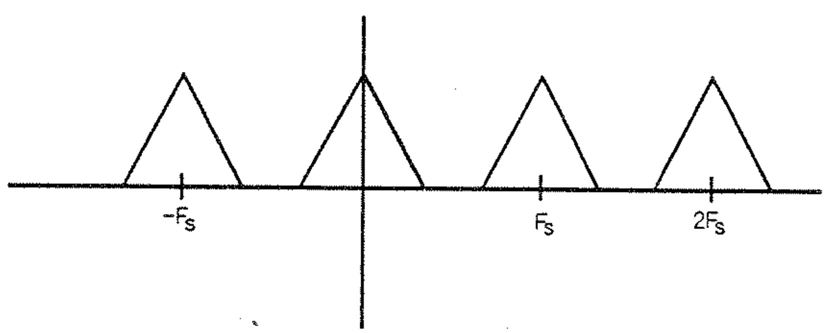


Figure 2-24. Sampling Causes Spectral Images To Repeat

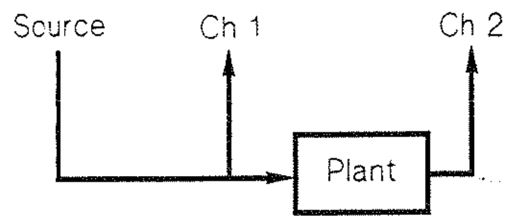


Figure 2-25. Measurement Setup: Plant Characterization

We will first characterize the plant, then make the mixed-domain measurement of the DAC-plant combination. Although it is not necessary to start with a plant characterization, it's useful when explaining the details of a mixed-domain measurement.

The measurement setup is shown in figure 2-25.

MEAS DISP FREQ RESP	Selects the frequency response function display.
SOURCE SOURCE LEVEL 5 V	Sets the analog source output level to 5 V. Random noise is the default source type.
SCALE Y FIXED SCALE 20, −75 dB	Sets the display scale to range from 20 dB to −75 dB.
AVG NUMBER AVGS?	Turns on averaging.
25	Sets number of averages to 25.
STABLE	Specifies stable averaging.
START	Begins the measurement.
X	Turns on the x-marker and moves it to the resonance peak. See figure 2-26.

Note that although the plant response has a general low-pass characteristic, a fairly significant resonance is present at around 17.5 kHz (see the marker values in figure 2-26). Energy from image frequencies above 15.625 kHz ($F_{sd}/2$) excite the resonance and are amplified, rather than attenuated.

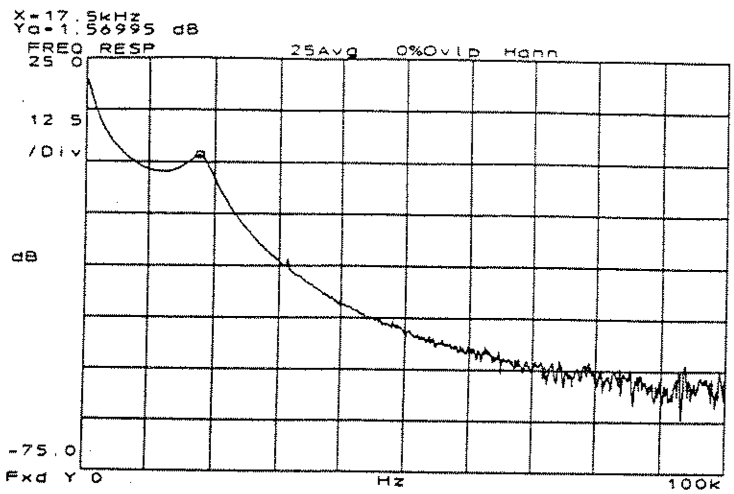


Figure 2-26. Plant's Frequency Response

Now we know the response of the plant. Next we will model the DAC that precedes the plant, and then combines the two responses to model expected response of the combination. The DAC can be modeled by a ZOH (zero-order hold), which has a frequency response shown in figure 2-27. To generate this response, press:

B **SYNTH** DOMAIN S **Z** POLYNOMIAL CLEAR TABLE	This series of key presses synthesizes a filter response with no loss or gain at any frequency; this is the filter characteristic of a piece of wire. Then we turn on the effects of a zero-order hold. The resulting response is due entirely to the ZOH sampling.
CLEAR TABLE	Clears the table of any previous values. (Preset doesn't clear it.)
EDIT NUMER#? ADD VALUE 1 ENTER SYNTH FCTN SAMPLE FREQ? 31.25 kHz RETURN RETURN	Edit numerator number? defaults to 1 if no entry is made.
CREATE TRACE 0 HOLD **ON** OFF Z DOMAIN	Synthesizes a trace in the z-domain with zero-order hold turned on in Trace B. We'll keep it in Trace B to compare with results displayed in A.
SCALE Y FIXD SCALE?	Set the y-axis scale to a 100 dB range.
25, −75 dB	See figure 2-27.

Note the general low-pass response. Also note the nulls at multiples of the sample frequency.

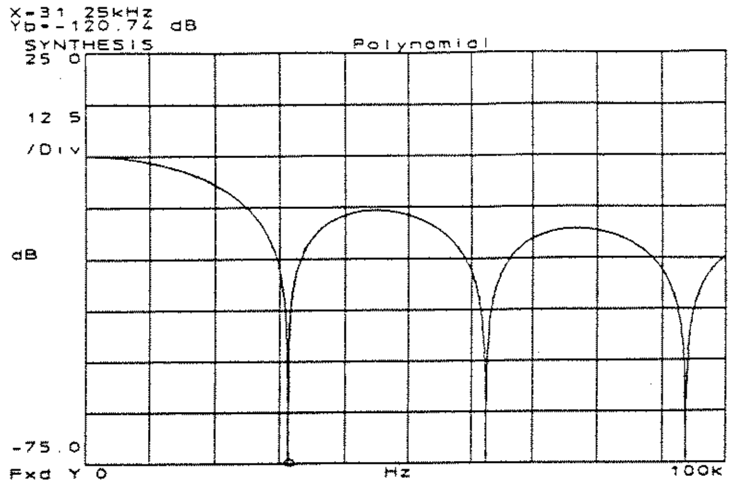


Figure 2-27. The Zero-Order Hold Response of a DAC

To see the effect the DAC response will have on the plant, press:

MATH MPY TRACE A	Multiplies the active trace (B) with Trace A. This response is the predicted response of the DAC-plant combination. (Recall that Trace A contained the plant FRF.)
X OFF	Turns of the x-marker. See figure 2-28.

We have an expected response. Next we will make the actual DAC-plant measurement and compare it with our predicted response.

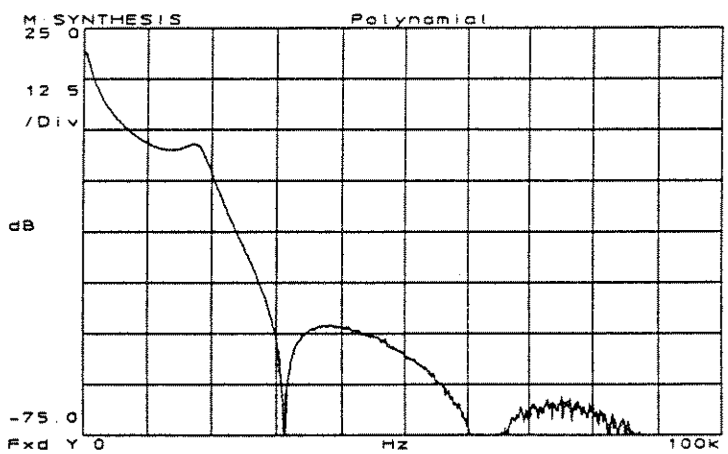


Figure 2-28. Predicted Response

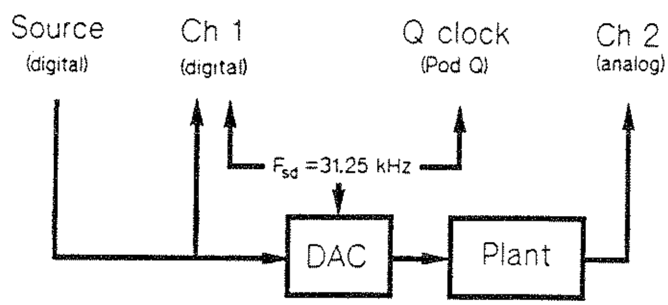


Figure 2-29. Mixed-Domain Measurement Diagram

The technique for making this mixed-domain measurement involves stimulating the DAC-plant combination with the digital source (we'll use random noise) and monitoring the analog response. This type of measurement involves connecting 2 clocks — F_{sd} is used as the Data Clock on Pod 1, and also as the Pod Q clock. See figure 2-29.

To set up this measurement, press:

INPUT CONFIG CHAN 1 AN DIG	Changes Channel 1 to digital. Note: default source is From Pod 1.
INTRFACE 1	
CHAN 1 CONFIG	Display the configuration on the screen. See figure 2-30

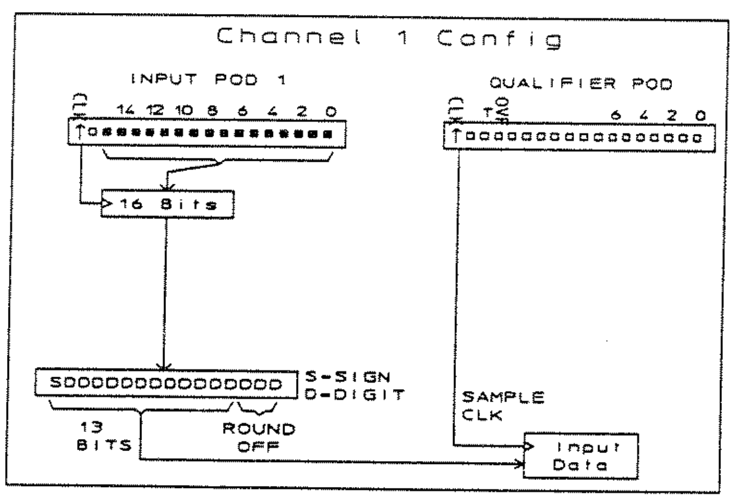


Figure 2-30. Channel 1 Configuration

SAMPLE CLOCK	
SAMPLE FREQ?	
31.25 kHz	
MIXED RATIO?	Since the same clock is used for both the Data Clock and Sample Clock, the
1 ENTER	Mixed Ratio is 1.
SOURCE	
SOURCE AN DIG	
INTERFACE	
SOURCE RANGE	
5 V	(the DAC full-scale level is 5V.)
RETURN	
SOURCE LEVEL	The measurement is noisier if you use larger values.
1 V	
A	Selects Trace A to show the response.
START	Starts the measurement.
SCALE	Sets the y-axis scale. See figure 2-31.
Y FIXD SCALE?	
20, − 10 dB	

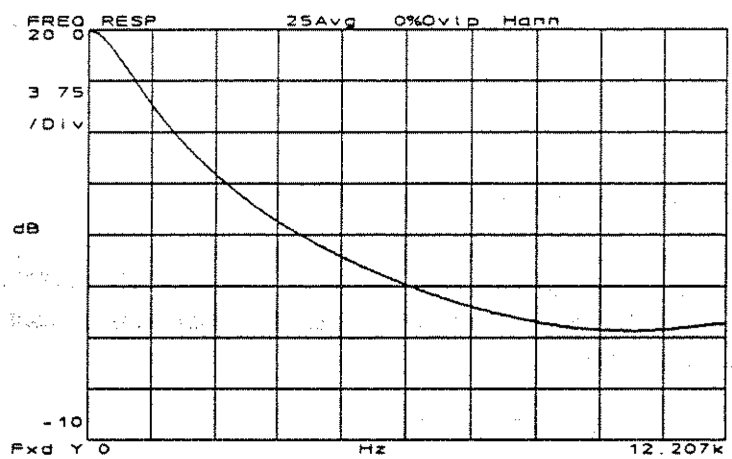


Figure 2-31. Mixed Domain: Ratio = 1:1

Since our sample frequency is 31.25 kHz, we only see up to $F_{sd}/2.56$, or 12.207 kHz. $F_{sd}/2$ is 15.61 kHz but calibration is specified only to $F_{sd}/2.56$, which corresponds to 801 lines (under the **WINDOW** hardkey). To view all the way to $F_{sd}/2$, select 1024 lines instead of 801. To compare with our synthesized measurement, we need to adjust the scaling:

A&B	Sets the scale for both axes.
SCALE	
X FIXED SCALE	
0, 12.207 kHz	$12.207 = 31.25 \div 2.56$
Y FIXED SCALE	
20, −10 dB	
FRONT BACK	Overlays the A and B traces for comparison.

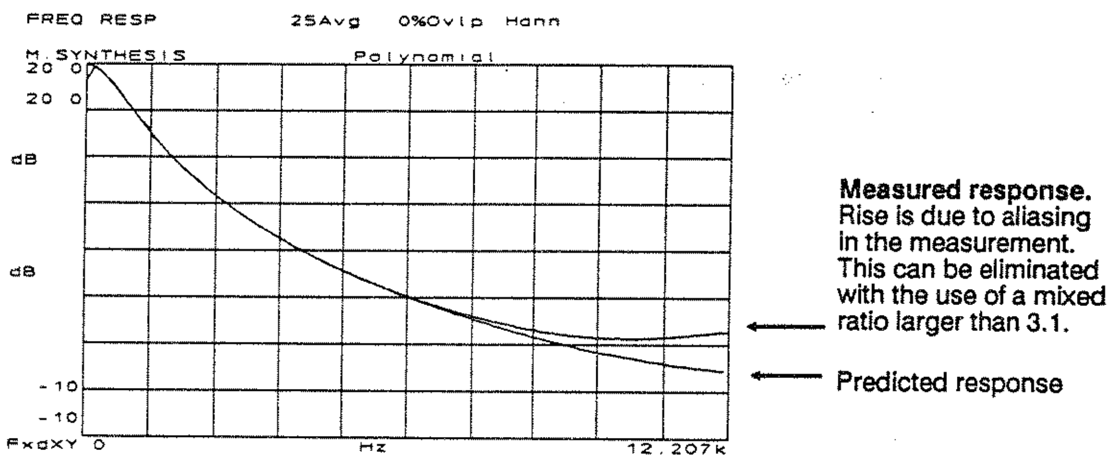


Figure 2-32. Comparing Measurement and Synthesis

Note the differences in these two traces. We should check to see if we're making an alias-free measurement. Refer to "Mixed Ratio" in chapter 3, or you can press **HELP/MIXEDRATIO** to get the same explanation on the display. To make an alias-free mixed-domain measurement, we can pick an analog sample frequency by

$$F_{sa} = F_{stop} + F_{top}$$

where:

F_{stop} = the highest frequency for which energy is present in the analog signal.
F_{top} = highest frequency to be measured; let's chose 35 kHz
$F_{sa} = 35\text{ kHz} + 62.5\text{ kHz} = 97.5\text{ kHz}$
F_{sa}/F_{sd} = minimum Mixed Ratio = $97.5\text{ kHz} / 31.25\text{ kHz} = 3.1$

Referring back to the predicted response, figure 2-28, signal levels are below 80 dB (from full-scale) at all frequencies above 62.5 kHz (even after the first null of the ZOH). So we will assume this is enough attenuation to consider the signal bandlimited. Thus $F_{stop} = 62.5$ kHz. See figure 3-13.

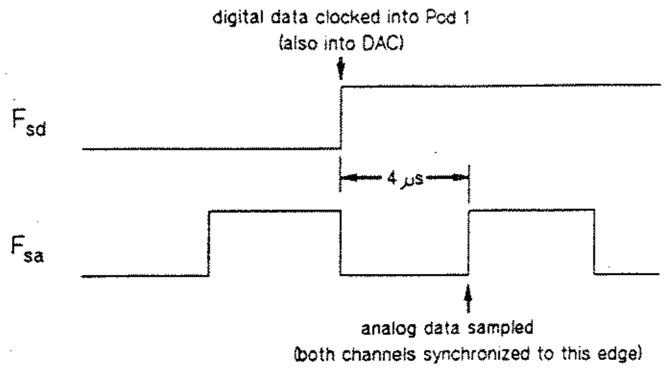


Figure 2-33. A Comparison of Clocks

It appears that our 1:1 ratio measurement is contaminated by aliasing. This is indicated by the rise in the response around 12 kHz due to the resonance of the plant aliasing into the measurement span. To make an alias-free measurement, we need a clock at least 3.1 times faster than F_{sd}. A clock at 4× F_{sd} (125 kHz) is available in the system. We connect this 125 kHz clock to Pod Q. The phase relationships of these clocks is shown in figure 2-33.

To configure the analyzer, press:

INPUT CONFIG INTERFACE 1 SAMPLE CLOCK MIXED RATIO 4 ENTER RETURN DATA CLOCK COMPUT DELAY	The clock polarity for Sample Clock is +; same as the Data Clock. Since the positive edges on 4 F_{sd} and F_s are out of phase, a phase ramp will appear in the data. We could select the – edge for Q clock, but to illustrate the use of computational delay, we will leave it +.
– 4 uS	A negative value is entered because F_{sa} "leads" 4 F_{sd}
UPPER/LOWER	Sets the display.
A	Selects Trace A to receive the measurement results.
START	Starts the measurement.
A&B **SCALE** X FIXDSCALE	Sets up the display scaling.
0, 48.828 kHz Y FIXED SCALE 20, -75 dB	Sets the frequency axis of both traces to match the digital measurement. (Remember, the predicted measurement was over a 100 kHz span.) See figure 2-34.

FRONT BACK See figure 2-35.

Figures 2-34 and 2-35 compare the actual and the predicted measurements. The noise at the higher frequencies is a signal-to-noise problem that could be reduced by changing to Swept Sine.

What have we accomplished? We made an alias-free measurement. Using the mixed-ratio feature allowed us to look past $F_{sd}/2$ and examine circuit behavior. The conclusion to be drawn from these measurements is that the DAC-plant combination is not a good anti-alias filter. Image frequencies are not adequately attenuated — they alias back below 15.625 kHz ($F_{sd}/2$). This causes distortion in the closed loop response.

To provide better alias rejection, two alternatives can be implemented:

- Design a dedicated anti-alias filter to proceed the ADC.
- Design a notch filter to attenuate image energy that could excite the resonance.

Summary

The mixed-domain measurement allowed us to measure the frequency response of a network that contained both analog and digital signals. Using the mixed-ratio feature allowed us to measure above the digital sample rate without the affects of aliasing. This was essential to determine the response of a portion of a sampled-data control system.

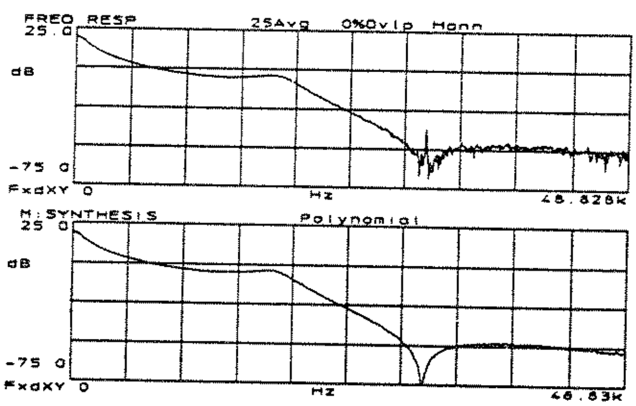


Figure 2-34. Upper/Lower Comparison

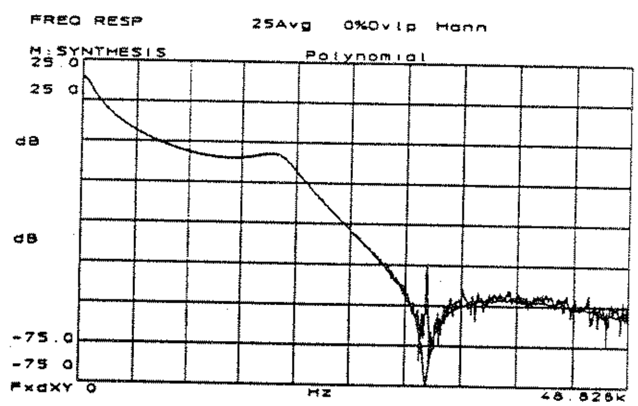


Figure 2-35. Front/Back Comparison

Chapter 3

Control System Methods and Models

3

Control System Methods and Models

This chapter develops some standard control systems models and measurement methods. More detail is provided in the three EDN articles in the last appendix. These methods and models are used in chapter 4 "Control System Tutorial."

General Model of a Control System

The standard model of a single-loop control system consists of the following:

- An input signal (r)
- A device, process, or "plant" to control
- A sensor to measure the response of the plant (c)
- A feedback loop and summation block to give an indication (b) of the result so that they may be combined with the original control signal to form an error signal (e)
- An error signal; that is sent to the plant such that c is driven to minimize the error between c and r. In figure 3-1, the plant's system function is G; the feedback system function is H.

The goal is to optimize closed-loop performance. This is done by modifying the open-loop elements. The ideal control loop output (c) tracks the input (r) perfectly in the time domain; the gain (output/input) is 1 and there is no phase lag between input and output.

Figure 3-1 is a generic block diagram of a control or "servo" system. In a real environment, there are external and internal disturbances that can affect the system's performance. Compensation is added to improve performance and ensure stability of an acceptable level. Figure 3-2 shows the compensation block with a transfer function of G_c. Many variations exist, out we can make assumptions:

- The control loop may be composed of mechanical devices and/or analog or digital electrical elements. In a digital system, summing junctions may not perform addition.
- Either no controller is necessary, or it is contained in the summation block.
- Either no compensation exists (figure 3-1) or it is contained in another block.

These systems have a closed-loop transfer function $\frac{C}{R} = \frac{G}{1+GH}$

Solving for the open-loop transfer function $\left(\frac{b}{e}\right)$: $b = (r-b)GH \qquad e = r - eGH$

$$\frac{b}{r} = \frac{GH}{1+GH} \qquad \frac{e}{r} = \frac{1}{1+GH}$$

The open-loop transfer function is $\frac{b}{e} = GH$

where: $G = G_c G_p$
G_c is the transfer function of the compensation network
G_p is the transfer function of the "plant" or process
H is the transfer function of the feedback network

For the derivation of the transfer function, see appendix B, *Control System Development Using Dynamic System Analyzers.*

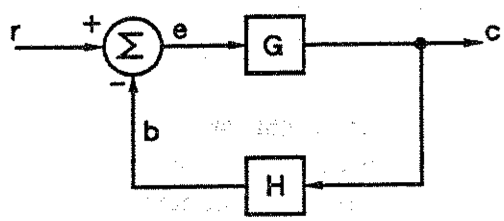


Figure 3-1. An Uncompensated Control System

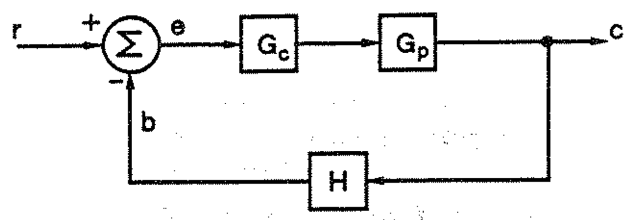


Figure 3-2. Compensation Added to Forward Path

Variations from the General Model

To analyze the performance of a control system, you must introduce a disturbance and monitor the effect. Some control systems have special requirements for testing and analyzing the results — this may limit how you approach setting up the measurement:

- Some systems have no access to the signals between blocks.
  - Printers and plotters that move a head around may have one chip that performs as a summing junction, compensation block, and feedback block.
- Sometimes the signals of interest are in a form that is difficult to measure.
  - If the plant is a motor, the signal that drives it may be pulse-width modulated.
  - In a switching power supply, the output signal may be a very large voltage.
  - When the error signal (e) is analog, it may be so small that the signal-to-noise ratio is poor.
- Some systems require a stimulus signal $r \neq 0$.
  - Testing disk drives requires that the disk be operating at all times; in this case so you cannot assume r = 0. This affects the math and your approach. However, if r(t) is uncorrelated to the stimulus signal, it can be treated as noise and averaging can be used to remove its effects.

Each situation can be solved by using variations of the general block diagram.

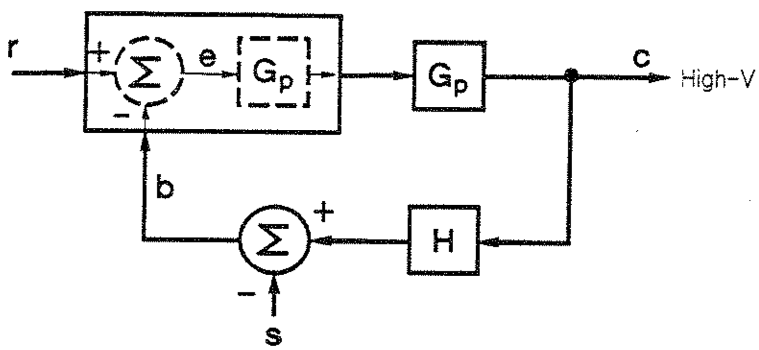


Figure 3-3. Possible Power Supply Block Diagram

Measurements

Here is a typical measurement sequence:

1. Draw a block diagram that represents your system, including signal path, types, and levels.
2. Work out the math that shows how to derive the open-loop response from the measurement.
3. Make the measurement.
4. Implement the HP 3563A math that transforms the measurement into the open-loop response.

This section covers steps 1 and 2 of this sequence. The examples covered here include:

- Finding the open-loop FRF using the reference input summing junction (measure b/e)
- Finding the open-loop FRF using another summing junction (measure y/z)
- Measuring the closed-loop FRF and calculate open-loop FRF (measure y/s)
- Measuring closed-loop FRF (V_{out}/V_{in}) and calculate open-loop FRF (measure c/r)

Measuring b/e

If you have access to signals at b and e, and the input node (r connection) is available, the best way to derive the open-loop response is to connect Channel 2 to b and Channel 1 to e (see figure 3-4) and display the frequency response function (referred to as either FREQ RESP or FRF). There is no math to do beyond selecting the FREQ RESP measurement (under **SELECT MEAS**) and display (under **MEAS DISP**). On the other hand, in a high-gain analog system this approach may not work, since e may be so small that it approaches the noise floor of the system.

If the system is completely digital, a slight variation of this method is preferred. The digital stimulus is connected to r and to Channel 1 (channel 1 can be connected to the digital source by pressing the FROM SOURCE softkey in the **INPUT CONFIG** menu). Channel 2 is connected to the digital b node. Then the b/r measurement can be used to calculate the open-loop response using the $\frac{T}{1-T}$ math function found in the **MATH** menu.

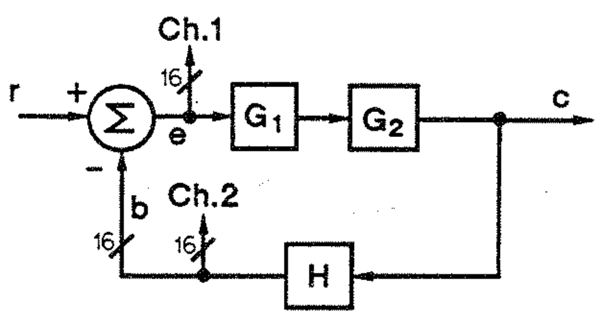


Figure 3-4. Measurement Method With Minimal Overhead

Solving for b yields: $b = -bG_1G_2H + rG_1G_2H$ from which we find $\frac{b}{r} = \frac{G_1G_2H}{1 + G_1G_2H}$

from which we can solve for the open-loop response: $G_1G_2H = \frac{b/r}{1 - b/r}$

which is of the form $\frac{T}{1 - T}$, a math function built into the HP 3563A.

Measuring y/z

See figure 3-5. A summing junction is used to add a stimulus signal (disturbance) to the loop. Solving for the open-loop frequency response, $\frac{b}{e}$:

$$e = -eG_1G_2H + r \qquad b = -bG_1G_2H + rG_1G_2H$$

$$\frac{e}{r} = \frac{1}{1 + G_1G_2H} \qquad \frac{b}{r} = \frac{G_1G_2H}{1 + G_1G_2H}$$

$$\frac{b}{e} = G_1G_2H$$

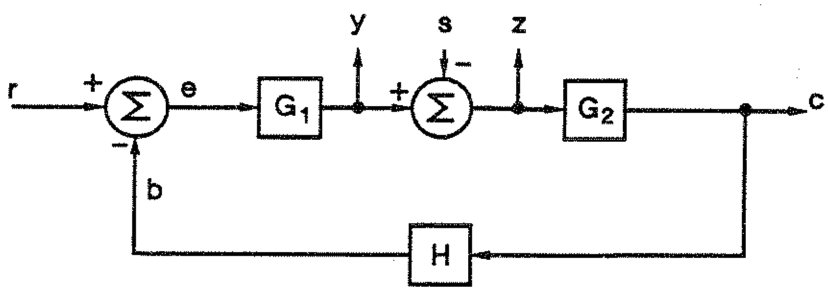


Figure 3-5. Control System Measurement Model

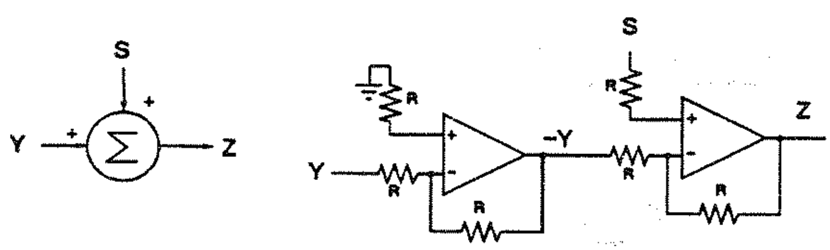


Figure 3-6. Summing Junction: Non-Inverting Inputs

Using the block diagram shown in figure 3-5, we solve for $\frac{y}{z}$ (assuming r very small or constant)

$$y = -yG_1G_2H + sG_1G_2H \qquad z = -zG_1G_2H - s$$

$$\frac{y}{s} = \frac{G_1G_2H}{1 + G_1G_2H} \qquad \frac{z}{s} = \frac{-1}{1 + G_1G_2H}$$

$$\frac{y}{z} = -G_1G_2H$$

This is almost identical to the open-loop transfer function. The only difference is the sign $\frac{b}{e} = -\frac{y}{z}$. The math necessary to derive the open-loop response simply negates the measured frequency response. Or, you can use a summing junction like that in figure 3-6.

Measuring y/s

Instead of measuring at y and z we can measure at y and s (connect Channel 1 to the source output). This reduces the number of connections to the circuit under test. However, it requires a calculation to obtain the open-loop frequency response. This measurement also demonstrates a math feature of the HP 3563A.

Earlier we found: $\frac{y}{s} = \frac{G_1G_2H}{1 + G_1G_2H}$; solving for the open-loop response: $G_1G_2H = \frac{\frac{y}{s}}{1 - \frac{y}{s}}$.

This is similar to the HP 3563A math feature $\frac{T}{1 - T}$ where T is the FRF displayed trace $\frac{y}{s}$.

To display the open-loop transfer function, we use the FRF measurement display to get $\frac{y}{s}$ and then use $\frac{T}{1 - T}$ in the math menu to do waveform math. Make sure $\frac{y}{s}$ is measured properly. At low frequencies the gain of $\frac{y}{s}$ should be very nearly 1 (0 dB) and the phase should be approximately 0 degrees (see figure 3-7). As discussed in the previous measurement example, it's easy to inadvertently pick up a 180° phase offset in the measurement results. Also, gain offsets can be picked up in the measurement. It's important to check the measurement results before math operations are performed on intermediate measurements. The math functions can also be used to "correct" a measurement containing phase and gain offsets.

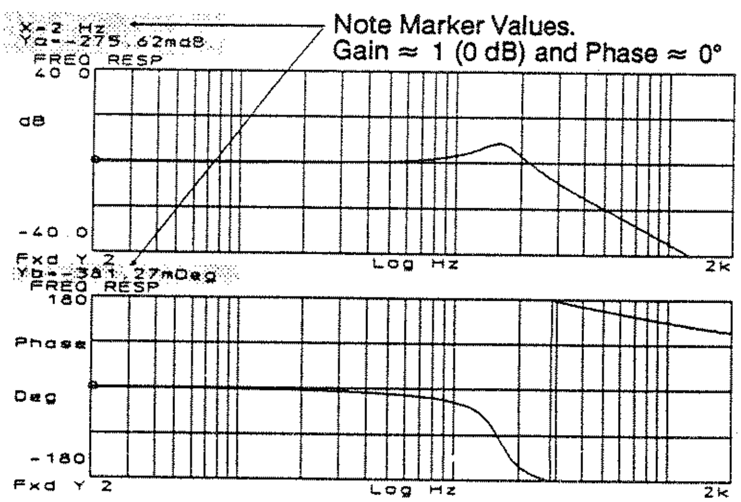


Figure 3-7. Magnitude and Phase Reality Check

Measuring c/r

Given the special case where H = 1, we can use another approach that doesn't require the addition of a summing junction. In this case, we can stimulate the circuit at r, measure r and c, display the FRF (c/r), and calculate the open-loop transfer function with waveform math.

From figure 3-8 we can show that $C = (R - CH)G$ and from this the FRF is

$$\frac{C}{R} = \frac{-G}{1 + GH}\bigg|_{H=1} = \frac{-G}{1 + G} \text{ then, solving for G in terms of } C/R, \quad G = \frac{-C/R}{1 + C/R}$$

can be accomplished with the following math: **SAVE RECALL**, SAVE DATA **1**, **MATH**, add **1**, div by SAVED 1, RECIP, NEGATE.

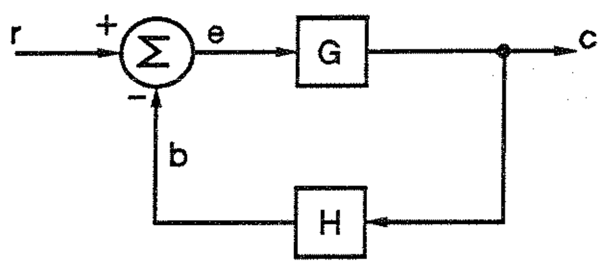


Figure 3-8. Simplest Measurement is c/r

Feedback Compensation

Compensation may also exist in the feedback loop. The system transfer function for figure 3-10 is

$$\frac{C}{R} = \frac{G_p}{1 + G_pG_c}$$

Since stability is mostly dependent on the pole location of the system's open loop transfer function, the roots of the denominator are critical determinants for stability. These are values that satisfy $1 + G_pG_c = 0$; where the open-loop gain approaches a value of 1 and the phase approaches $-180°$. Note that the compensator system function appears in the denominator whether the compensator is in the feedback or forward path of the system.

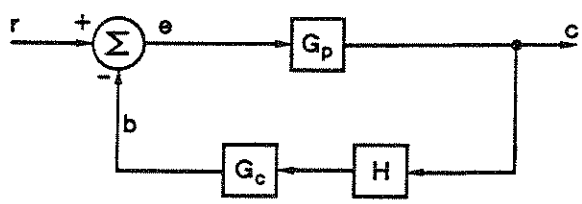


Figure 3-9. Compensation in the Feedback Path

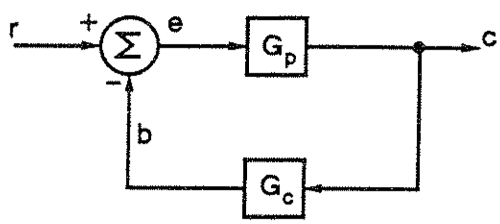


Figure 3-10. Feedback Compensation with H = 1

Mixed Domain (Digital/Analog) Control System

The analog compensator in figure 3-10 can be replaced with a digital compensator as shown in figure 3-11. The analog output signal is sampled with a sample-and-hold circuit, and then converted to a digital signal with a analog-to-digital converter (ADC). The digital compensator is a digital filter. Then the compensator's output is converted back to an analog signal with a digital-to-analog converter (DAC) and, possibly, a reconstruction filter (a low-pass filter to reduce images that occur on either side of multiples of the sampling frequency F_s). There are many possible variations of control loop design, with more or less of the loop being composed of digital elements.

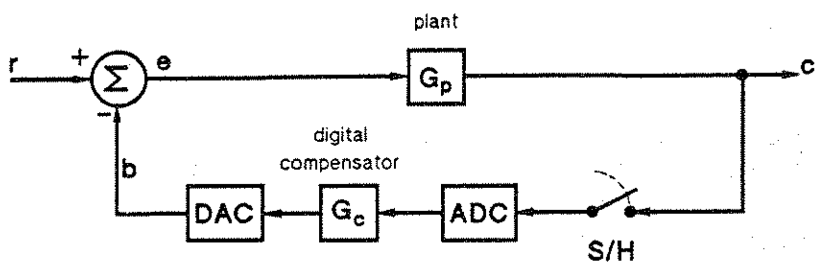


Figure 3-11. Digital Compensation in an Analog System

Mixed-Domain Measurements and Analysis

Now that we have created a system composed of analog and digital circuits, it's time to consider measurements that span continuous-time and sampled-time data.

Mixed Ratio

The Mixed-Ratio value specifies the ratio between the analog and digital sample rates in a mixed-domain measurement. Mixed ratio can be used to avoid measurement aliasing and to measure beyond half the digital sampling frequency ($F_s/2$) on the analog channel. The default mixed-ratio value is 1. Values must be an integer between 1 and 512.

For mixed ratios other than one, two clocks are used while making a mixed-ratio measurement. One clock is used for the digital channel, and another faster clock is used for the analog channel. Only one clock is needed if the ratio is 1:1. The clock for the digital channel is connected to one of the input pod clocks (CHAN 1 CLOCK or CHAN 2 CLOCK). The digital sample rate, F_{sd}, is entered as the sample frequency for the measurement.

The clock for the analog channel is connected to either the Pod Q clock (POD Q CLOCK) or External Sample (EXT SAMPLE). (If the ratio is 1:1, the analog rate is the same as the digital rate. The digital clock is selected for the sample clock.) The analog sample rate, F_{sa}, is not entered. The analyzer determines F_{sa} from the MIXED RATIO, which is an integer multiple from 1 to 512 of F_{sd}.

Example

Digital Sample Rate = F_{sd} = 1 kHz (1 kHz clock connected to CHAN 1 or CHAN 2 CLOCK)

MIXED RATIO = 4

Analog Sample Rate = F_{sa} = 4 kHz (4 kHz clock connected to POD Q or EXT SAMPLE)

It's important to understand the phase relationship between the two clocks in a mixed ratio measurement. Clocks that are skewed result in a phase ramp in any frequency-domain measurements involving phase. However, this ramp can be corrected by manually entering a delay value corresponding to the clock skew, using the COMPUT DELAY softkey (**DATA CLOCK** menu).

Since the analog data is sampled at a higher rate than the digital data, the digital data is internally resampled to match the analog rate. This resampling introduces zero amplitude samples between the original sample points.

Note

The number of non-zero points in the digital time record decreases as the mixed ratio value increases. A mixed ratio of 512 creates a digital time record with only four non-zero samples.

This effect can be reduced by narrowing the measurement span using the FREQ SPAN softkey in the **FREQ** hardkey menu. Decreasing the span by a factor of two increases the number of non-zero points by a factor of two.

Note

If viewing the source in a mixed-domain measurement and the mixed ratio is 1, the measurement process inserts zeros in the *filtered time record.* If the source waveform already contains zeros, zero insertion (from the measurement process) may cause all of the waveform to be 0.

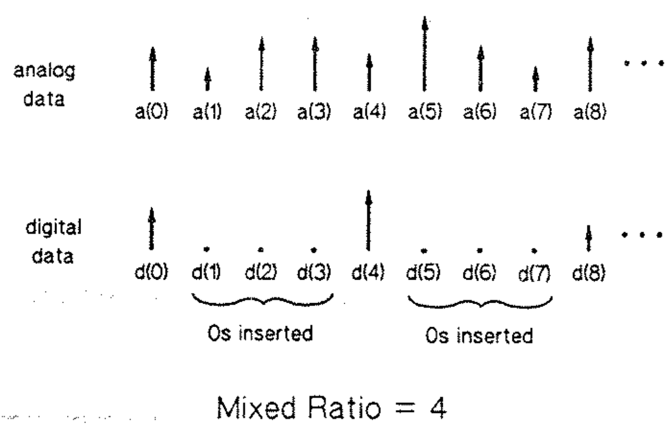
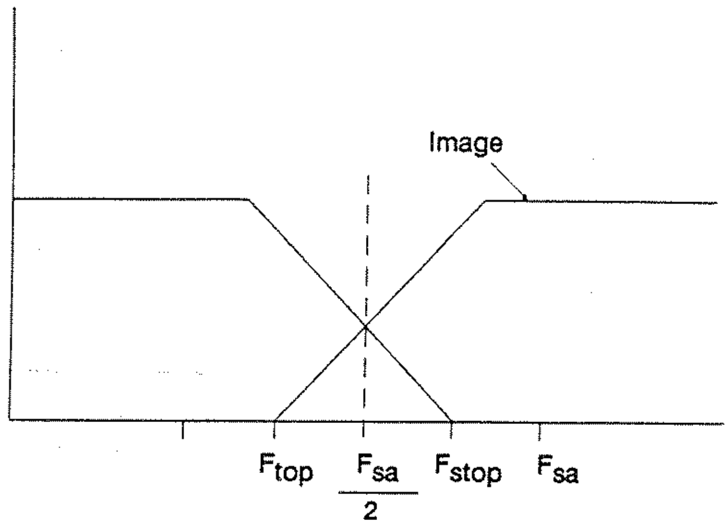


Figure 3-12. The Mixed Ratio's Effect on Data



$F_{top} = F_{sa} - F_{stop}$

$F_{sa} = F_{top} + F_{stop}$

Figure 3-13. Calculating F_{sa}

The appropriate analog sampling frequency, F_{sa}, is based on the range of frequencies to be measured and on the potential for aliasing. The following formula is used to select F_{sa}:

$$\frac{F_{sa}}{2} = \frac{F_{stop} - F_{top}}{2} + F_{top}$$

This simplifies to:

$$F_{sa} = F_{stop} + F_{top}$$

Where:

F_{sa} = Analog Sample Frequency

F_{stop} = The highest frequency for which energy is present in the analog signal

F_{top} = The highest frequency be accurately measured

This formula places $F_{sa}/2$ halfway between F_{stop} and F_{top}. Energy in the span from $F_{sa}/2$ to F_{stop} will alias into the span from F_{top} to $F_{sa}/2$. This leaves the span from 0 to F_{top} alias-free.

The HP 3563A's anti-alias filter (on the analog channel) bandlimits the analog signal to 156 kHz. F_{stop} can always equal 156 kHz. If the analog signal is externally band-limited to a frequency lower than 156 kHz, that lower frequency should be used instead.

Example

This example explains how to measure the sin(x)/x response of a digital-to-analog converter (DAC) operating at 44 kHz. The highest frequency to be measured is 47 kHz.

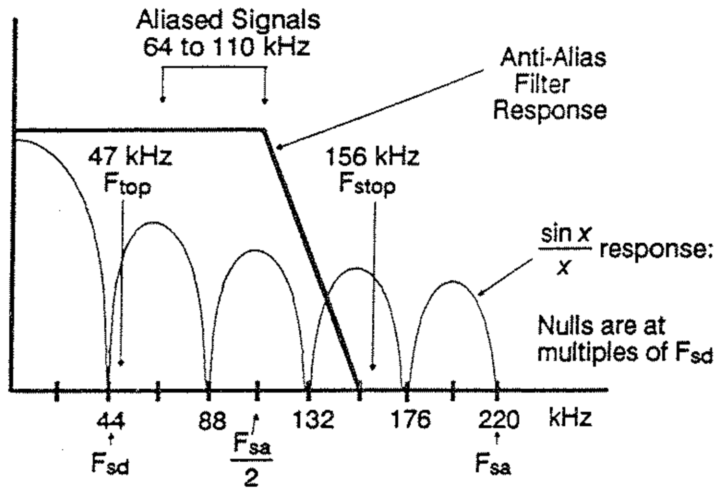


Figure 3-14. Digital-to-Analog Converter Response

Channel 1 is in digital mode and is connected to the DAC input. Channel 2 is in analog mode and is connected directly to the output of the DAC. There is no low pass filter on the DAC output so the output of the DAC is not band-limited. This example, therefore, specifies 156 kHz for F_{stop} and 47 kHz for F_{top}.

Use the formula, $F_{sa} = F_{stop} + F_{top}$, to determine the minimum analog sampling frequency.

$$F_{sa} = 156\ \text{kHz} + 47\ \text{kHz} = 203\ \text{kHz}$$

The analog-to-digital ratio (mixed ratio) is: $\frac{F_{sa}}{F_{sd}} = \frac{203\ \text{kHz}}{44\ \text{kHz}} = 4.61$

The mixed ratio must be an integer; 4.61 rounds up to 5. (Always round up.) Then:

1. Set the sample frequency, F_{sd}, to 44 kHz.
2. Connect a 44 kHz clock to the Channel 1 Input Pod Clock input.
3. Set MIXED RATIO to 5.
4. Connect a 220 kHz clock ($F_{sd} \times$ Mixed Ratio which is 44 kHz × 5) to either the EXT SAMP or the POD Q CLOCK input.

$\frac{F_{sa}}{2}$ is 110 kHz (220/2). Anything in the frequency span between 110 kHz and 156 kHz will fold back into the span between 65 kHz and 110 Khz. Energy above 156 kHz is removed by the HP 3563A's anti-alias filters and does not alias.

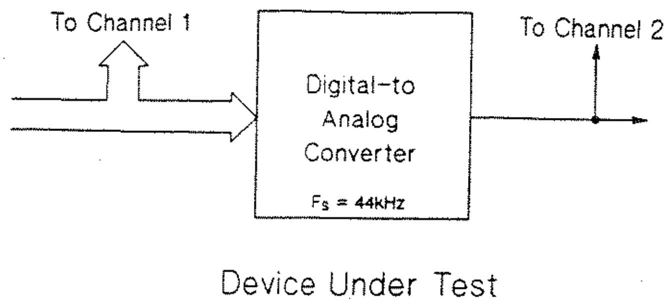


Figure 3-15. DAC Frequency Response Measurement

In the example, the maximum analyzer span is 85.94 kHz in 800-line mode.

$$\text{MAX FREQ SPAN} = \frac{800}{1024} \times \frac{F_{sa}}{2}$$

equivalent to: $\frac{F_{sd}}{2.56} \times \text{Mixed Ratio} = \frac{44\text{ kHz}}{2.56} \times 5 = 85.94\text{ kHz.}$

To avoid the data above 64 kHz distorted by aliasing, the center frequency and span should be set so the 800 lines of resolution are concentrated between 0 Hz to 64 kHz.

The analyzer's frequency spans are based on the F_{sa}. As the mixed ratio increases, the maximum displayed span increases. As the analyzer's span increases, the resolution for the digital spectrum, 0 to $F_{sd}/2$, decreases.

Note

If the math operations COMPRESS, EXPAND, or EXTRACT are used with mixed ratio, the mixed ratio value must be a power of 2.

Chapter 4

Control System Tutorial

4

Control System Tutorial

This chapter presents measurement and analysis exercises that demonstrate the use of the HP 3563A in the analysis and design of control system compensation. Methods and models that are basie to these excercises were developed in chapter 3.

This tutorial shows how to add compensation to a known control system to improve performance specifications. The *objective* is to implement compensation with a digital filter. The following steps calculate analog compensation parameters. These are then converted to the z-domain for implementation. Design calculations and program steps are shown below:

1. Measure the step response of the control system (time domain).
2. Measure the closed-loop response of the control system (frequency domain).
3. Calculate the open-loop response; find gain and phase margins.
4. Curve fit to get a pole-zero model of the system.
5. Design analog compensation (work shown).
6. Synthesize proposed compensation .
7. Combine compensation response with system response.
8. Check gain and phase margins of the combination response.
9. Use math features to synthesize compensated step response.
10. Adjust compensation design as necessary, and repeat steps 4 through 9.
11. Curve fit the synthesized compensation design in the z domain (discuss transform options).
12. Design digital filter.
13. Measure FRF of digital compensator (digital-digital) to verify implementation.
14. Put digital compensator in control system feedback loop.
15. Measure closed-loop response.
16. Calculate the open-loop response; find gain and phase margins.
17. Compare with results obtained in step 8.
18. Do a final check of the system step response.

Key-Press Conventions

The key presses are called out as follows:

HARDKEY
  1st-level softkey
    2nd-level
      3rd-level
	A hardkey is any key other than the eight keys on the right side of the display. Hardkeys usually display a menu of softkey items adjacent to the eight display keys. Softkeys may change the configuration or call other softkey menus.
	Each of the listed keys should be pressed. The amount of indent in the keypress listing indicates relative level in the menu structure.

Testing an Uncompensated System

Step Response

The step response is the measured reaction of the control system to a step change in the input. The HP 3563A source has a step output that we'll use here to measure a step response. Figure 4-1 shows the connections you should use between the analyzer and the device under test (in this case, a simple control system block diagram). Note that our model's summing junction inputs are both inverting.

Take a measurement

This is an analog-in/analog-out measurement using an analog source. To configure the analyzer from the turn-on or preset state, press the following keys:

SOURCE LEVEL 1 V	Sets the source level to 1 volt. (Your system may require a different value.)
SOURCE TYPE MORE TYPES STEP	Selects the source type "step."
FREQ FREQ SPAN 20 kHz	Sets the frequency band to limit the response to frequencies between 0 Hz and 20 kHz. This lengthens the time record from 8 ms (at span = 100 kHz) to 40 ms.
RANGE CHAN 1 RANGE 1.25 V CHAN 2 RANGE 1.75 V	Sets range on both channels. These values were determined by the source level setting. Channel 1 is connected to the source, so its range can be set to about the same value. Since some overshoot is likely on Channel 2, we chose a range setting somewhat higher than that for Channel 1.
SELECT TRIG SOURCE TRIG	Selects the source as the trigger. You must select something other than FREE RUN for trigger delay to work.

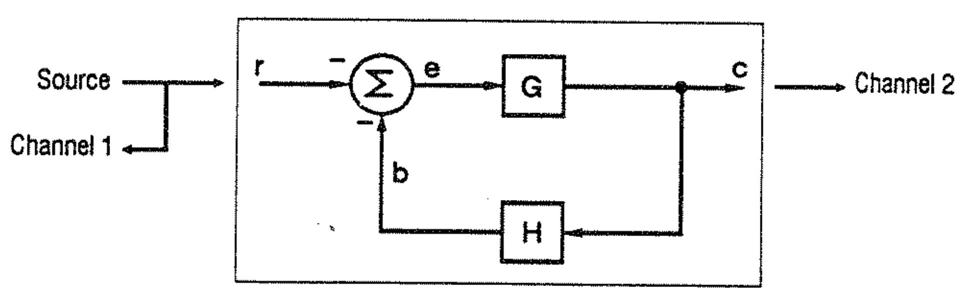
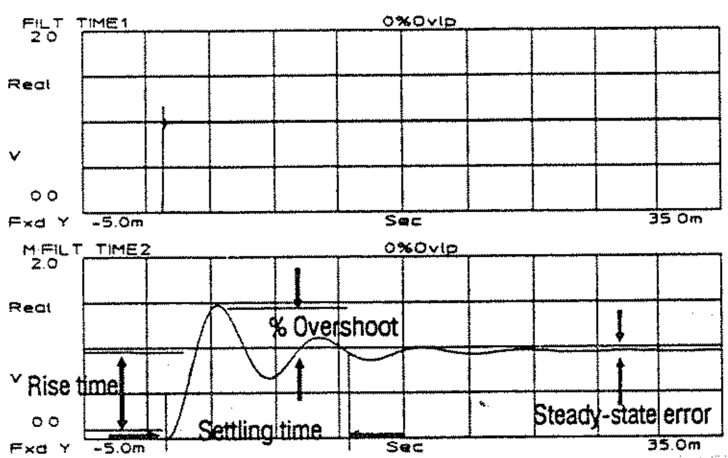


Figure 4-1. Step Response Analyzer Hook Up

TRIG DELAY – 5 mSec	Delays the trigger (of both Channels 1 and 2) 5 milliseconds, which allows easy viewing of the rising edge.
UPPER/LOWER	Displays two traces, one above the other.
A **MEAS DISP** FILTRD INPUT TIME REC 1 **B** TIME REC 2	This sets the display to show the Channel 1 time record on Trace A and the Channel 2 time record on Trace B.
AVG STABLE (MEAN) NUMBER AVGS 1 ENTER	This step is used to stop the measurement process after one step. To repeat the measurement, press **START**.
A&B **SCALE** Y FIXED SCALE 0,2 V	Selects both traces for the next configuration step. Sets the vertical scale for both traces; the bottom of the scale is 0 V and the top is 2 V.
START	Starts the measurement.
B **MATH** NEGATE	Activates only Trace B for the following math operation. Negates the active trace. This is necessary to offset the inversion caused by the summing junction reference input (r). The step response should appear approximately as it does in figure 4-2.



The top trace shows the source step signal as measured by Channel 1. Because the signal passes through the Channel 1 anti-aliasing filter, the response has some ringing.

The lower trace shows the step response of the device under test. Points of interest include :

1. Settling time
2. Rise time
3. % Overshoot
4. Steady-state error

(1, 2, and 3 should be measured on a normalized trace.)

Figure 4-2. Source Step (top) and Step Response

Use Markers to Measure Performance

Normalize Trace

X Markers Knob	Turns on the x-marker on the active trace (in this case, trace B). Rotate the markers knob clock wise to *position the marker dot at the extreme right edge of the trace.* This places the marker in the steady-state region. The difference between this value and the signal level of the source step is the steady state error.
MATH DIV **MARKER VALUE**	This step divides Trace B by the marker value (assumes Trace B is active). The **MARKER VALUE** key is a hardkey in the ENTRY block. Now the trace has been normalized to an amplitude of 1.

Find Settling Time

Y HOLD Y CENTER Markers Knob	Turns on the y-marker. The marker appears at the center of the screen, and reflects the normalized value. Rotate the knob to *set the value of ΔY at approximately 100 mV.* This sets a horizontal band or zone (± 5%) in which the response is said to be "settled," the response stays within the boundaries.
SINGLE	Expands Trace B to fill the display area. This makes it easier to view. (X MRKR SCALE can be used to "zoom" in more on fine details.)
X Markers Knob HOLD X LEFT Markers Knob	Displays the x-marker menu. Rotate the knob to *position the x-marker at approximately time = 0.* Turns on the X "band" markers. Rotate the knob to *position the Δx- marker at the point where the step response stays within the y-band markers.* You can now read the settling time value may directly from the Δx- marker readout.

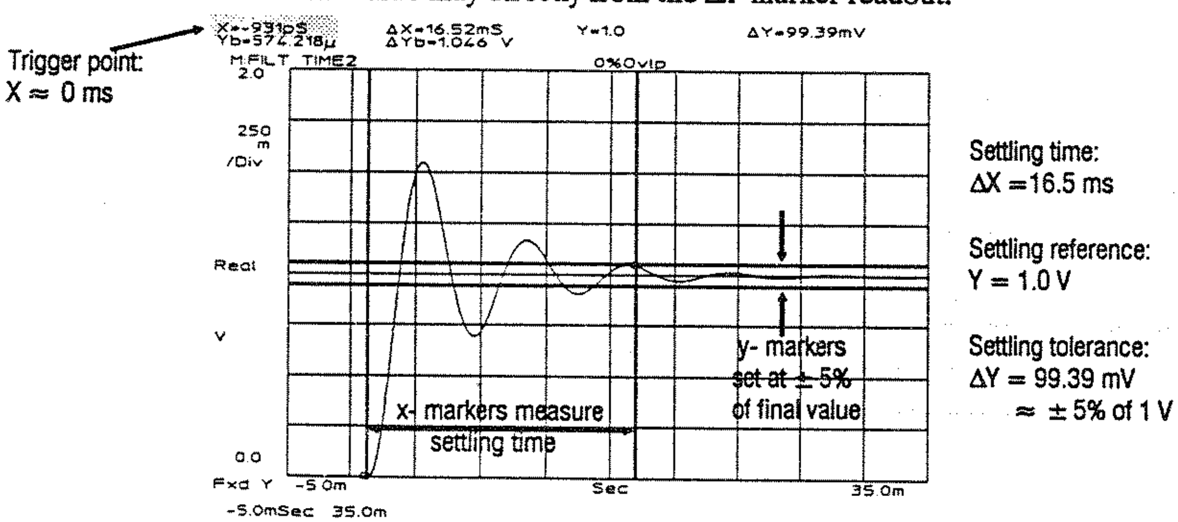


Figure 4-3. Measuring Settling Time

Find Rise Time

X OFF **Y OFF**	Turns off both sets of markers. Turning the markers off and on again is a good way to turn off the "band" markers.
Y	Turns on the y-marker again.
Markers Knob	Rotates the markers knob to *position the y-marker line at 100 mV.*
HOLD Y LOWER	Turns on the y-band or Δ (delta) marker.
Markers Knob	Rotates the markers knob to *set the* Δ*y-marker at 800 mV.* The y-markers are now positioned at the 10% and 90% levels of the step function.
X	Turns, on the x-marker again.
Markers Knob	Rotate the knob to *position the x-marker* at the 10% value of the rise time (*where the lower y-marker crosses the trace*). Turn on the ΔX marker.
HOLD X LEFT	
Markers Knob	Turn the knob to position the ΔX at the point where the upper y-marker crosses the trace. The rise time value may now be read directly from the ΔX marker readout. See figure 4-4.

Find % Overshoot

X OFF **Y OFF**	Turns off the x- and y-markers.
X	Turns on the x-marker again. This appears at the largest value of the trace — the peak of the overshoot. Since the trace has been normalized, the marker value readout, minus one, is the percent overshoot. From a visual inspection of figure 4-4, you can see that the overshoot is about 55 percent.

Step Response Summary

We have used the source output "step" to stimulate the control system and measured some of its time-domain characteristics. Next we will examine its frequency-domain characteristics.

It's good practice to display the stimulus signal on one trace as a reference. The digital filters will affect the measurement if the span is set too low.

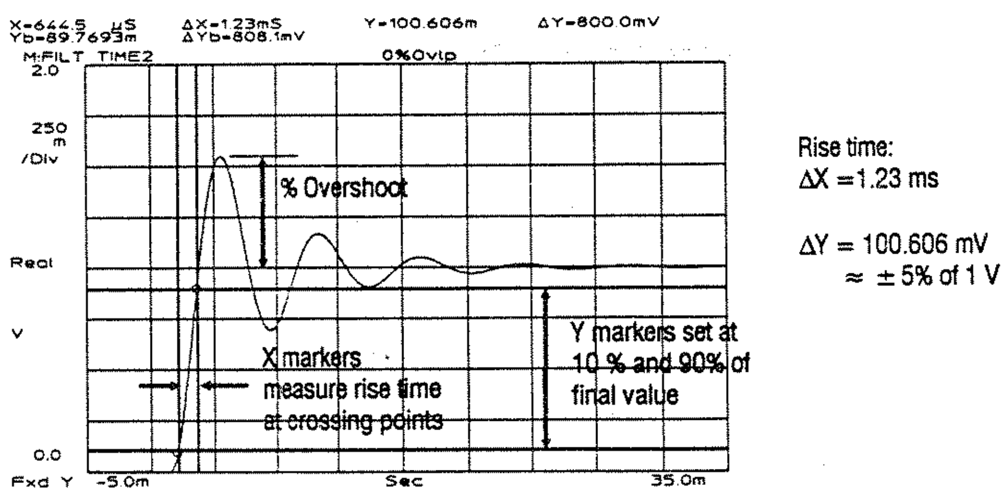


Figure 4-4. Measuring Rise Time

Swept Sine FRF

The system in figure 4-5 consists of a controller (G_1), plant (G_2), and feedback circuit (H = 1). The open-loop transfer function is modeled as $GH(s) = \frac{K \times 10^6}{(s + 884)(s + 72)(s + 16)}$ (in Hz). We're going to use frequency-domain analysis to characterize its open-loop response. The goal is to add digital compensation to improve the response and stability.

First, we need to verify the model. Then we will design an analog compensator. Finally, we will convert the analog compensator to a digital compensator. These procedures determine the system stability by measuring the gain crossover, gain margin and phase margin. The desired frequency-domain specifications are:

- Gain crossover: 265 Hz ± 20 Hz
- Gain margin: 12 dB ± 2 dB
- Phase margin: 40° ± 5°

To test the system, we need to "disturb" it by injecting a signal between the controller and the plant. This is done by adding a summing junction and connecting the HP 3563A source. Connect the analyzer to the system as shown in figure 4-5 and perform the following key presses. We are going to measure y/z as discussed in "Control Systems Methods and Models."

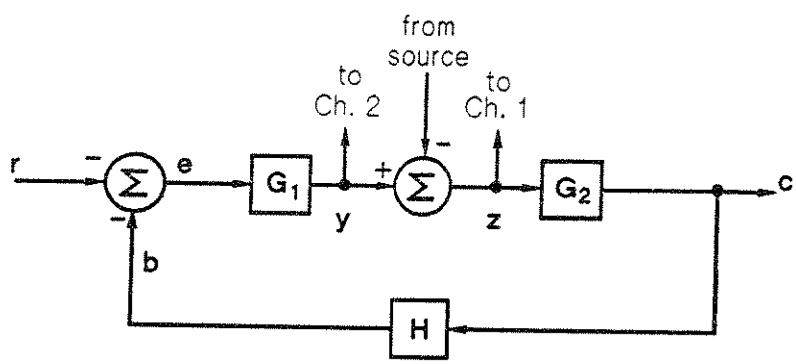


Figure 4-5. Connecting Analyzer to Control System

Make the Measurement

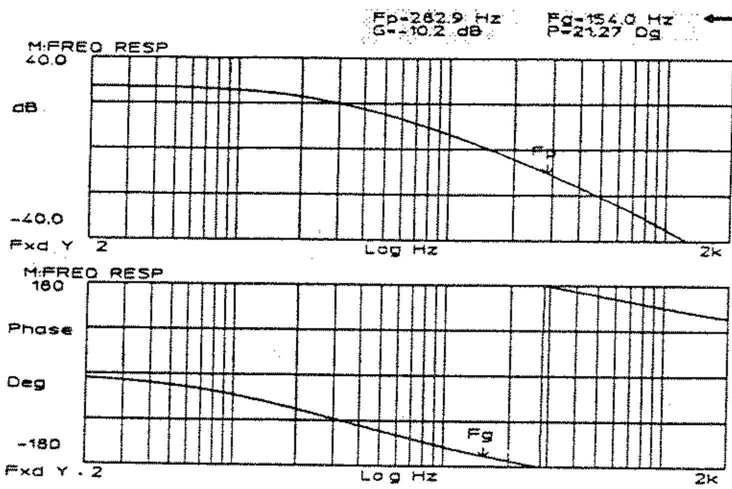
MEAS MODE SWEPT SINE	Selects swept sine as the measurement mode.
SOURCE SOURCE LEVEL 300 mV	Sets the source level to 300 mV. Note that "300" are key presses in the numeric key pad and "mV" is a softkey that terminates the data entry. (This value varies with the system under test.)
AVG AUTO INTGRT?	Selects automatic integration time and allows entry of variance threshold. This allows the analyzer to select the optimum integration time. If you make no entry, the default value (5%) is used.
FREQ RESLTN <u>AU</u> FIX START FREQ 2 Hz	Sets the resolution to automatic. This allows the best measurement. Sets the start frequency to 2 Hz. The default span is three decades, so the stop frequency is 2 kHz.
A&B **MEAS DISP** FREQ RESP	Displays two traces in upper/lower format; selects both traces for next step. Changes the displayed data from power spectrum (the default) to Frequency Response Function (sometimes called FRF).
A **SCALE** Y FIXD SCALE – 40,40 dB	Selects the A (upper) trace for configuration. Displays the Scale softkey menu. Begins data entry for fixing the scale. Sets the scale for Trace A to extend from – 40 to +40 dB. (The comma and – are hardkeys in the Entry block, beneath the key labeled " 3.")
B **COORD** PHASE	Selects the B (lower) trace for configuration. Selects phase display for the currently selected trace.
START (wait)	Starts the measurement process. Wait for the sweep to finish. This measurement should take about four minutes.
A&B **MATH** Negate	Selects both traces for the next configuration step. Negates the traces to compensate for the inverting input of the summing junction. This changes the B (phase) trace (shifts data 180 degrees) and makes both traces display data in memory (note the M:FREQ RESP). See the discussion in "Control Systems Methods and Models" to learn why this calculates the open-loop response.

Find the Gain and Phase Margin Values

X OFF — Make sure the X markers are off, since they limit the margins calculation.

SPCL MARKER
- Marker Calc
  - Gain & Ph Mgn

Calculates the gain and phase margins and displays the data above the active trace (at top of display if both traces active; frequencies are always displayed at the top of the screen). See figure 4-6.



From the marker readouts we get:

Gain margin (G) = − 10.2 dB
the magnitude below 0 dB
(gain = 1) at the frequency at
which phase = − 180°
this occurs at F_p = 283 Hz

Phase margin (P) = 21.3°
the phase above − 180° at the
frequency at which gain = 1

Gain crossover = 154 Hz
the frequency at which gain = 1

Figure 4-6. Open-Loop Frequency Response Bode Plot

Measure Gain and Phase at Desired Gain-Crossover Point

Our specification goal is to have a gain-crossover frequency of 265 Hz. At this frequency, we also want a phase margin of 40° ± 5°. We need to know the phase value at 265 Hz for the compensator design problem we will soon encounter. The compensator gain will be determined by the gain required to achieve gain *crossover* at 265 Hz. The gain *margin* does not occur at 265 Hz; it is measured at a frequency where the phase is – 180°. So next we will measure the existing gain and phase at 265 Hz.

X	Assuming both traces are selected, this turns the x-marker on both traces.
X VALUE?	Activates the data entry to position the marker.
265 Hz	Positions the x-markers at 265 Hz. The display now appears as in figure 4-7.

Marker measurements show the following values:

- Gain = – 9.15 dB
- Phase = – 178°

From these numbers we can see that, at 265 Hz, the compensator will need 9.15 dB of gain and 38° of phase shift (180 – 178 = 2; 40 – 2 = 38).

Gain at 265 Hz: – 9.15 dB

Phase at 265 Hz: – 178°

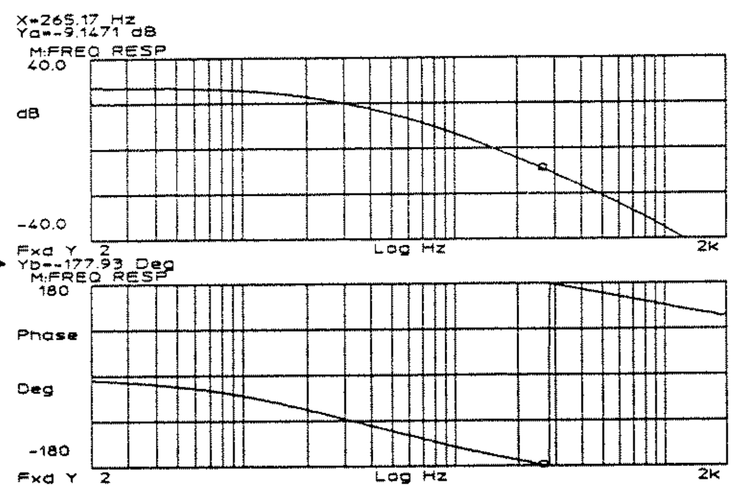


Figure 4-7. Gain and Phase Readings at 265 Hz

Analyzing Test Results

In the preceeding steps we measured a gain margin of 10.6 dB and a phase margin of 22.2°. We also measured a gain crossover of 150 Hz. Before designing the compensator, we need to verify that our control system model is correct. We can do this by curve fitting to derive the poles and zeros of the response and comparing them to our model.

Curve Fitting

Curve fitting is "the adjustment of the parameters of a mathematical model of a physical system, so the performance of the model matches the measured performance of the physical system in some optimal manner." (See the glossary in appendix C, *Curve Fitting in the HP 3562A*.) The result of curve fitting is a table of poles and zeros. There is much to know about curve fitting; more than can be covered in a tutorial such as this in a reasonable amount of time. Appendix C reviews the most important aspects of curve fitting and lists specific steps to obtain the best results.

To perform a curve fit, press the following keys:

X OFF	Turns off the x-marker so that it doesn't limit the range of data used.
CURVE FIT A & B TRACES NUMBER POLES 5 ENTER NUMBER ZEROS 5 ENTER	We select the curve fitting to be performed on traces A and B, instead of the last measurement, because the measurement is not the open-loop response. Then we enter a limit of poles and zeros to use. Since we know the transfer function, we already know what poles and zeros to expect. Setting limits like this can reduce the calculation time substantially.
CREATE FIT START FIT	This step starts the curve fitter running.
(wait)	After starting the fit, wait for the message "Fit Complete" to appear in the lower-right corner of the display. It took about one minute to curve fit the trace in figure 4-8. The resulting curve appears as the lower trace. Note that the curve fit stopped at 190 Hz.

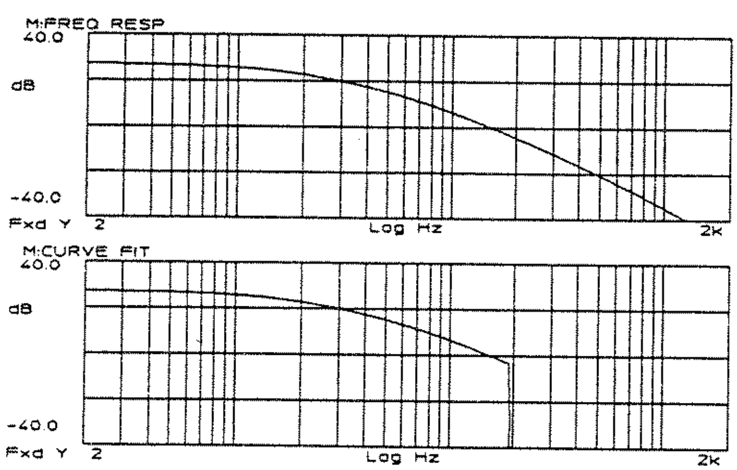


Figure 4-8. Magnitude and Curve Fit Results

Weighting Function

The curve-fitting algorithm in the HP 3563A uses a *weighting function*. This is a function along the frequency axis that "weights" the error in the curve fit so that some regions are given more influence to determine the final quality of the fit. This emphasizes peaks and valleys and de-emphasizes regions of the spectrum that have poor *coherence*. Weighting values range from 0 to 1 and may be edited. The weighting function is calculated automatically when curve fitting is performed (it may also be user-defined to emphasize a particular span of data). To view the weighting function used for this curve fit exercise, press:

A	Selects Trace A to display Weighting Function.
CURVE FIT FIT FCTN EDIT WEIGHT VIEW WEIGHT	This series of key presses displays the weighting function in Trace A.
A&B	Selects both traces so the x-marker will appear on both traces.
X	Turn on the x-markers.
Markers Knob	Rotates the knob to position the marker at the curve fit discontinuity. This displays the weighting function as shown in figure 4-9. The markers are set at the point where the curve fitter stopped using measurement data. Note that the weighting value is 0.001 and the frequency at that point is about 190 Hz. *The curve fitter doesn't use data weighted < 1 mUnit.*

Coherence is a function that represents the amount of output signal power that can be attributed to the input signal. Its value ranges from one (1) which is perfect coherence, to zero (0) which is no coherence (any value less than .9999 should not be considered "good"). Coherence is a function of variance, which (normally) is derived from multiple measurements that occur during averaging (in one of the FFT measurement modes, like linear resolution).

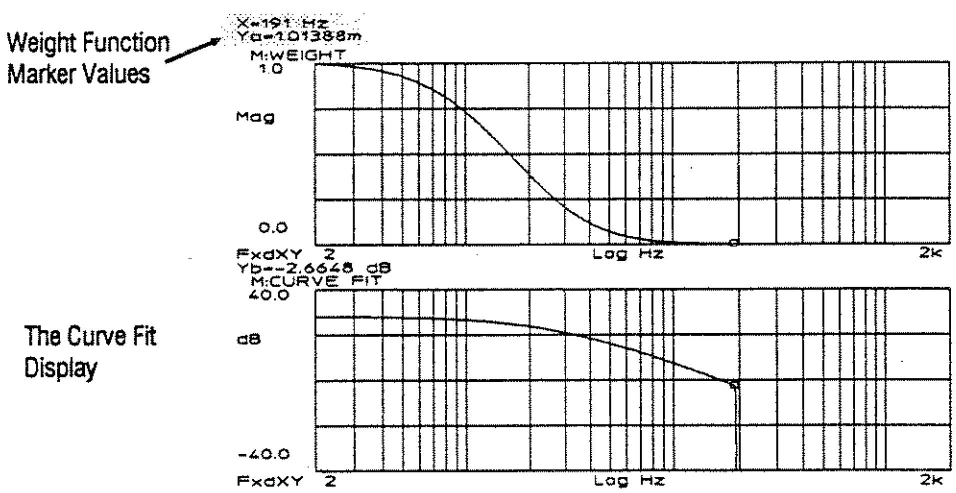


Figure 4-9. Magnitude and Weighting Displays

When the measurement mode is swept sine, there are no multiple measurements with which to calculate variance. Instead, a three-point moving average is used. This results in an approximation of the standard coherence results. For this reason, coherence data is not as reliable when determining the weighting function in swept sine measurements as in linear or log resolution measurements. A more thorough examination would include FFT analysis, which allows the use of averaging. This would yield better coherence data for the curve fitter. For a technical definition of coherence, see appendix A.

Poor data from the curve fitter is often the result of giving it bad data. To learn how to get good results from the curve fitter, see the "Curve Fitting Check List" in appendix E, *Curve Fitting in the HP 3562A*.

The result of curve fitting is pole-zero data in the curve fit table. To view the curve fit table, press:

CURVE FIT
  EDIT TABLE This displays the table shown in figure 4-10.

The table in figure 4-10 shows that the curve fitter found three poles and three zeros. The transfer function on page 4-6 shows three poles. So the curve fitter found the poles — but it also found some unexpected zeros.

S Curve Fit

	POLES S	ZEROS S
1	-885.136	-9.48019k
2	-72.6239	6.46715k±j 8.73598k
3	-15.9257	

Time delay=0.0 S Gain=21.2μ Scale=1.0

Figure 4-10. Curve Fit Table

Next we'll test the significance of the zeros. To do this, we copy the poles and zeros in the curve fit table to the synthesis table, remove the zeros, synthesize a spectrum from the remaining poles, and compare the results with the original open-loop response. First, we'll display data in the A and B traces:

A **MEAS DISP** FREQ RESP	These key presses put the frequency response measurement back in Trace A (the top trace).
MATH NEGATE	These key presses changes the sign of the measurement data to compensate for the inversion of the source signal at the summing junction. See figure 4-5 and discussion of measuring y/z in chapter 3. Don't expect the trace to change; this only affects phase and we are currently displaying gain. (The math affects the complex data from which the displayed data is generated.)
B **CURVE FIT**	We want the synthesized trace to appear in Trace B. Displays the Curve Fit softkey menu.
FIT FCTN FIT → SYNTH	Copies the curve fit data to the synthesis table.
SYNTH POLE ZERO EDIT ZERO# (select a zero) DELETE VALUE DELETE VALUE SYNTH FCTN GAIN FACTOR 23.744 EXP 6 ENTER RETURN RETURN CREATE TRACE S DOMAIN **FRONT BACK**	Displays the synthesis softkey menu. Of the three forms that we could deal with the curve fit data, select the pole/zero format. We want to delete zeros, so we select the "edit zeros" softkey. The pole and zero components are on lines which are numbered at the left. Select a component with the up/down arrow key, the data entry knob, or by entering the line number with the keypad. Selected components are bright. We can't remove the zeros without putting their magnitudes back in the numerator. The original gain factor was 21.2×10^{-6}. We should multiply this by the zeros' magnitudes; $9480.19 \times (6467.15^2 + 8735.98^2)$. The resulting gain factor (23.744×10^6) is then entered in place of the original. Now we have all the data in place. Pressing the Create Trace softkey displays a softkey menu which allows us to select which domain to use. Pressing the S Domain softkey starts the synthesis which displays its results. Then we superimpose the synthesized trace with the open-loop response for a critical comparison of the two, as shown in figure 4-11.

Frequency Response Summary

The synthesized trace (without the zeros) is a very close match to the open-loop response that we curve fit. To further examine the difference, you could divide one trace from the other and look at the difference. Running the curve fit again with the number of zeros constrained to zero allows the fitter to place the poles even more accurately. The results appear in figure 4-12. There is very little variation in the placement of the poles compared with the original table in figure 4-10.

Next, we will design the compensation. Then we will synthesize it, combine it with our measured open-loop response, and look at the performance characteristics.

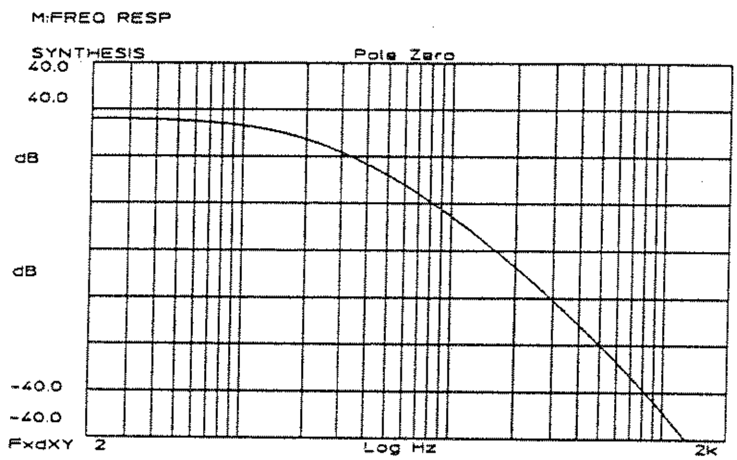


Figure 4-11. Measured and Synthesized Traces Overlaid

S Curve Fit

POLES 5	ZEROS 0
1 -883.35	
2 -72.6159	
3 -15.9258	

Time delay=0.0 S Gain=23.6M Scale=1.0

Figure 4-12. Curve Fit Table; Zeros Constrained to 0

Designing the Compensation

Now that we have verified that the model is accurate and have collected some performance data, we can design a compensator to improve performance to the new specifications. Our measured data compares to the frequency-domain specifications as follows:

Parameter	Goal Specification	Measured
gain crossover	265 Hz ± 20 Hz	150 Hz
gain margin	– 12 dB ± 2 dB	– 10.6
phase margin	40° ± 5°	22.2°

Our design problem requires the characteristics of a phase-lead compensator. Figure 4-13 shows the circuit configuration used to implement a phase-lead compensator. Figure 4-15 shows the response of a phase-lead compensator.

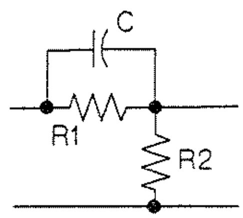


Figure 4-13. Compensation; a Phase-Lead Network

$$G_c = \frac{V_o}{V_i} = \frac{R_2}{R_2 + \frac{1}{1/R_1 + 1/X_c}} = \frac{R_2}{R_2 + \frac{R_1 X_c}{R_1 + X_c}} = \frac{R_2 X_c + R_2 R_1}{X_c(R_2 + R_1) + R_2 R_1} \text{ where } X_c = \frac{1}{SC}$$

substituting for $X_c = \frac{1}{SC}$ then $G_c = \dfrac{\frac{R_2}{SC} + R_2 R_1}{\frac{1}{SC}(R_1 + R_2) + R_1 R_2} = \dfrac{SCR_2R_1 + R_2}{SCR_2R_1 + (R_1 + R_2)} = \dfrac{S + \frac{1}{R_1 C}}{S + \frac{R_1 + R_2}{R_2 R_1 C}}$

and we let $T = R_1 C$ and $\alpha = \frac{R_2}{R_1 + R_2}$ (α is the dc attenuation of the network)

then the transfer function works out to be $G = \dfrac{S + \frac{1}{T}}{S + \frac{1}{\alpha T}}$ one pole at $\frac{1}{\alpha T}$ and one zero at $\frac{1}{T}$

From figure 4-14 the phase angle at $\omega = \omega_m$ is $\sin \varphi_m = \dfrac{\frac{1-\alpha}{2}}{\frac{1+\alpha}{2}} = \dfrac{1-\alpha}{1+\alpha}$ so $\alpha = \dfrac{1 - \sin \varphi_m}{1 + \sin \varphi_m}$

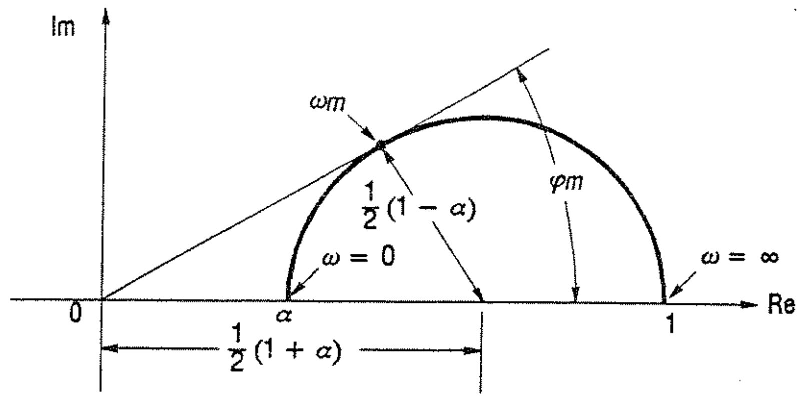


Figure 4-14. Polar Plot of a Phase-Lead Network

From the measurement at 265 Hz the maximum phase: $\varphi_m = 38°$ so $\alpha = \dfrac{1 - 0.6157}{1 + 0.6157} = 0.2379.$

Figure 4-15 shows the Bode diagram of a phase-lead network. The break frequencies occur at $\omega = 1/T$ and $\omega = 1/(\alpha T)$. ω_m is the geometric mean of the two break frequencies, so:

$$\log \omega_m = \frac{1}{2}\left[\log \frac{1}{T} + \log \frac{1}{\alpha T}\right] \text{ and } \omega_m = \frac{1}{T\sqrt{\alpha}}$$

So the *zero location* $\dfrac{1}{T} = 2\pi\, 265 \sqrt{\alpha} = 2\pi\, 265 \sqrt{.2379} = 2\pi\ (129.2 \text{ Hz})$ radians

The *pole location* $\dfrac{1}{\alpha T} = 2\pi \dfrac{129.2}{.2379} = 2\pi\ (543.3 \text{ Hz})$ radians

The compensation gain (magnitude of the transfer function) at 265 Hz is $20 \log \sqrt{\alpha} = -6.2$ dB. The original gain reading at 265 Hz was -9.15 dB. The gain needed to make the gain crossover occur at 265 Hz is $9.15 + 6.2 = 15.35$ dB. 15.35 dB converts to a *gain factor = 5.85* (solving for x where: $15.35 \text{ dB} = 20 \log x$).

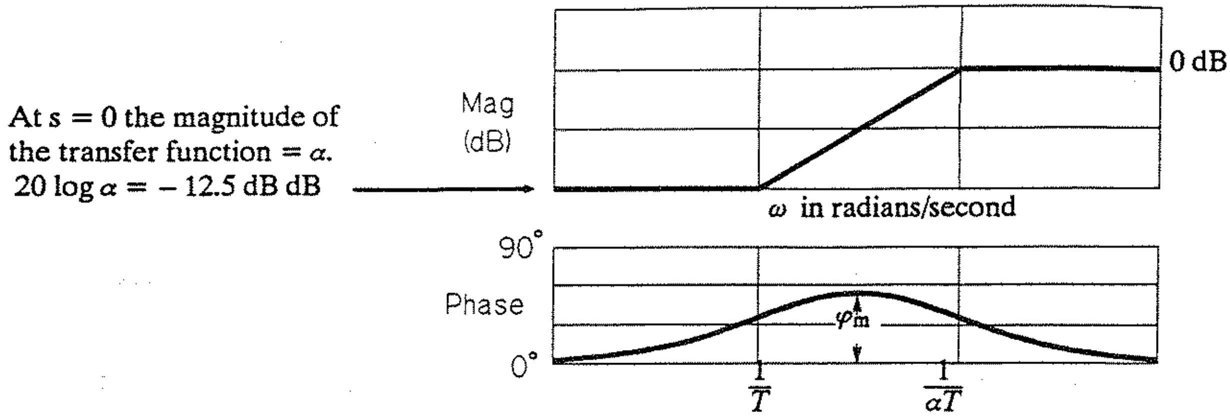


Figure 4-15. Bode Diagram of a Phase-Lead Network

Compensator Design Summary

Our design has a pole at – 543.3 Hz, a zero at – 129.2 Hz, and a gain factor of 5.85. To combine it with the measured response we can:

- Synthesize the compensator by itself and multiply the trace with the open-loop response trace to get the response of the combination of the two.
- Add the data to the synthesis table (which already contains data for the measurement) and synthesize the combination trace.

Checking the Design: Frequency Domain

We can synthesize the compensator and use trace math to multiply the compensator's synthesized response with the open-loop response. This gives us the open-loop response we would expect if we built the compensator, added it to the system, and measured it.

Synthesize the Compensator Response

The last steps we took had the open-loop response in Trace A, and the synthesized trace of the curve fit with the zeros removed in Trace B.

The traces were overlaid in the front/back display format. We want to enter the compensator design values in the synthesis table, synthesize it, look at it and then examine it.

B	Select sTrace B.
COORD PHASE	Sets the coordinates of the active trace to be phase.
UPPER LOWER	Selects the display format having two traces displayed one above the other.
A&B	Selects both traces for the following configuration steps.
SYNTH	Selects synthesis menu.
POLE ZERO	Selects data format.
CLEAR TABLE CLEAR TABLE	Clears old values from table. Must press key twice to clear the table.
EDIT POLE# ADD VALUE	We're going to add a pole to the table.
– 543.3 Hz	Adds a pole at – 543.3 Hz.
EDIT ZERO# ADD VALUE	Next, we add a zero.
– 129.2 Hz	Adds a zero at – 129.2 Hz.
SYNTH FCTN GAIN FACTOR	Next, we add the gain factor.
5.85 ENTER RETURN	Adds a gain factor of 5.85.
RETURN	
CREATE TRACE S DOMAIN	Creates a trace in the s domain.
(wait)	Waits for the trace to be synthesized. When it is complete its Bode diagram is displayed (because of the trace configuration steps done before synthesis).
SCALE Y AUTO SCALE	Auto-scales the results so they are easier to examine.
X X VALUE? 265 Hz	Positions the x-markers at 265 Hz so we can check the results of the design exercise. See figure 4-16. Compare it with figure 4-15.

From figure 4-16 we can see the compensator design parameters we need (compare this with the Bode diagram in figure 4-15). The phase (bottom trace) peaks at 265 Hz with a maximum value of 38°. The gain runs from about 3 dB at low frequencies to 15 dB at 2 kHz, with 9.1 dB at 265 Hz.

Next we will use trace math to combine this trace (the compensator's synthesized frequency response) with the control system's measured response. This yields the response of the control system with the compensator installed.

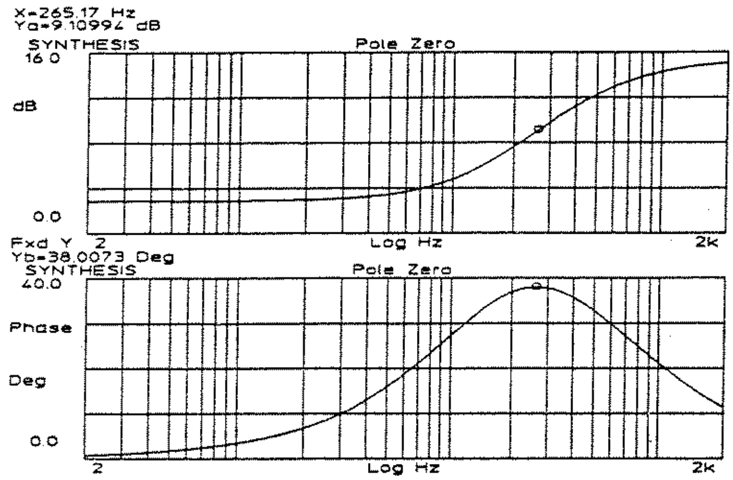


Figure 4-16. Synthesized Compensator Response

Combine Compensator Response with System Response

Set Up the Display Configuration

A MEAS DISP FREQ RESP	Puts the system measurement in Trace A.
MATH NEGATE	Changes the sign of the measurement data to compensate for the inversion of the source signal at the summing junction. See figure 4-5 and discussion of y/z in chapter 3. (If your conversion math is complicated, store the open-loop response in one of the 5 data registers after the first conversion, and recall it.)
B COORD MAG (dB)	Changes the lower trace to display magnitude of the synthesized compensator response.
FRONT BACK	Changes the display format so that the two traces overlay each other.
A&B SCALE Y FIXD SCALE? 40, – 40 dB	Selects both traces. Changes the vertical scale to range from – 40 dB (bottom) to +40 dB (top). The display now appears as shown in figure 4-17.

Do the Math

B MATH MPY? TRACE A	Selects Trace B as first argument and to receive results of the math. Selects the math menu. Selects math operation; "multiply (active) trace data by . . ." Selects Trace A as the second argument.

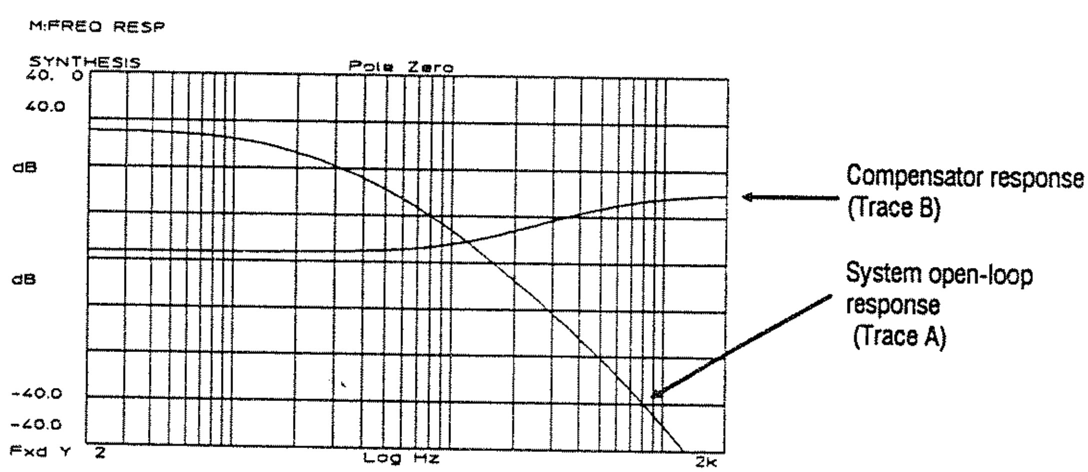


Figure 4-17. Compensator and System Responses Overlaid

Analyze Frequency Domain Results

X OFF — Make sure the x-markers are off; they limit the margins calculation.

SPCL MARKER
MARKER CALC
GAIN PH MGN — Uses the marker calculation to measure the gain and phase margins of the response of the active trace; in this case, Trace B, which is the combined responses of the system and compensator. Results appear in figure 4-18.

Taking numbers from the marker readout, the results are:

Parameter	Goal Specification	Results
Gain crossover	265 Hz ± 20 Hz	267 Hz
Gain margin	– 12 dB ± 2 dB	– 12.6 dB
Phase margin	40° ± 5°	39.8°

So it appears that this design almost exactly matches the goals for this system in the frequency domain. Next we shall examine the results in the time domain.

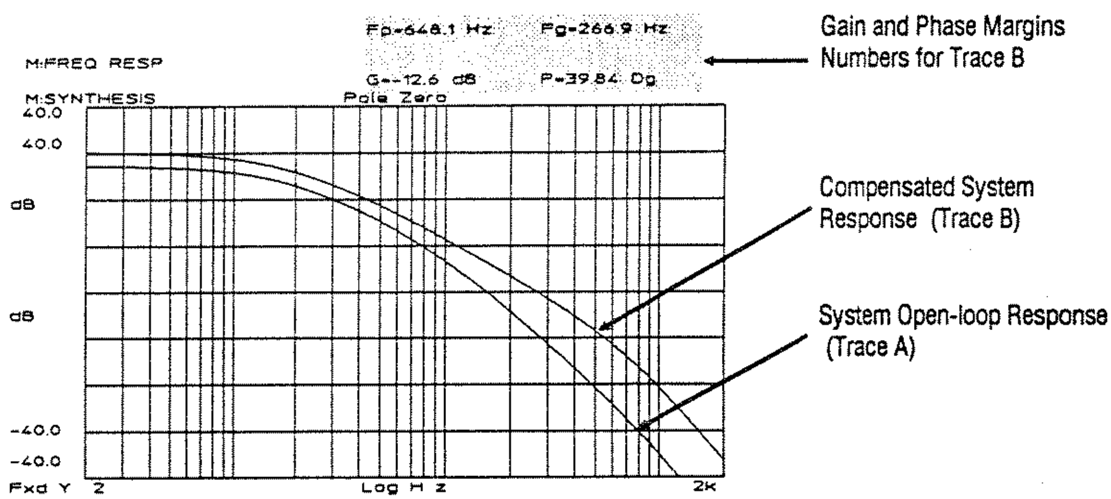


Figure 4-18. Compensated and Uncompensated Traces

Checking the Design: Time Domain

To check our design in the time domain, we will add the compensator's pole, zero and gain factor data to that of the measured response. We will then synthesize the results with the analyzer configured as it was when we did the original step function measurement. The results are frequency-domain data that can be converted to time-domain with the inverse FFT (FFT^{-1}).

Synthesize the Compensated System Response

We'll add the poles of the system response to the compensator data in the synthesis table. Then we will synthesize a trace.

MEAS MODE LINEAR RES	Selects the linear resolution measurement mode.
FREQ FREQ SPAN 20 kHz	Selects the frequency range from 0 Hz to 20 kHz.
A	Selects Trace A as the active trace.
SINGLE	Selects the display format to be one large grid showing only the active trace.
SYNTH POLE ZERO EDIT POLE# ADD VALUE – 883.35 Hz – 72.6159 Hz – 15.9258 Hz	Adds the poles from the previous curve fit of the control system's open-loop response (see figure 4-12).
SYNTH FCTN GAIN FACTOR 1.38 EXP 8 ENTER RETURN RETURN	Changes the gain factor to $23.6 \times 10^6 \times 5.85 = 1.38 \times 10^8$. This is the gain factor after we took out the zeros times the compensator's design value (see discussion with figure 4-11).
CREATE TRACE S DOMAIN	Creates trace in the s-domain.
SCALE Y AUTO SCALE	Selects auto-scaling of the vertical axis. The display now appears as shown in figure 4-19.

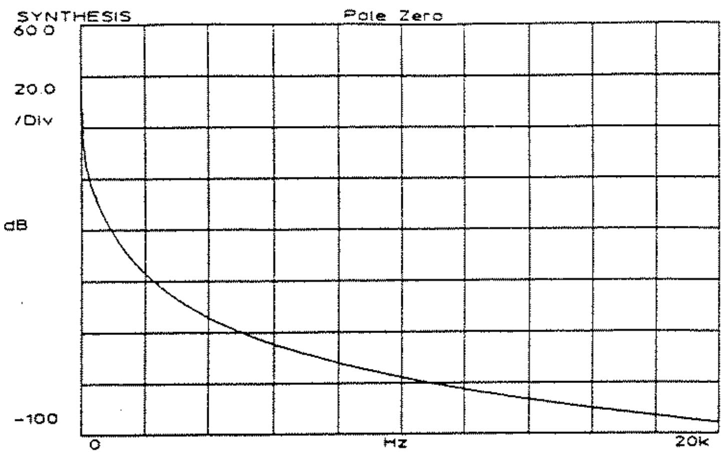


Figure 4-19. Synthesized Response in Linear Resolution

Use Trace Math to Generate Step Function

To get the step response from the synthesized (compensated) system open-loop response we will:

1. Convert the open-loop response to the closed-loop response.
2. Perform the inverse FFT.
3. Integrate (yields the step response).

Then we will normalize the response and measure the settling time, rise time, and overshoot.

SAVE RECALL
SAVE DATA#
1 ENTER

Saves the synthesized response in data register #1. (There are five registers.)

MATH
ADD
1 ENTER
RECIP
MPY
SAVED 1
NEXT
NEXT
FFT^{-1}
RETURN
INTGRT INIT=0

Converts the open-loop response to the closed-loop with math.

Adds 1 to the trace data; result becomes new trace data.
Divides 1 by trace data; result becomes new trace data.

Multiplies trace data by data saved in register #1 (the original trace).
What we have done so far is: $\frac{T}{1 + T}$
The inverse FFT converts FRF to impulse response.

Integrates the impulse response to get the step response.

X
Markers Knob

Now turn on the x-marker and rotate the knob to position it at the *extreme right side of the screen.*

Normalize the Trace

MATH DIVIDE	Divides the active trace by the marker value.
MARKER VALUE	Now the trace is normalized.
SCALE Y FIXED SCALE 0, 2 ENTER	This sets the scale so that the middle rule on the graticule is 1.

Find the Settling Time

Y HOLD Y CENTER Markers Knob	Turns on the y-marker. Rotate the knob until the value of ΔY *is approximately 100 mV.* This sets a horizontal band or zone (± 5%) in which the response is said to be "settled" when it stays within the boundaries.
X Markers Knob HOLD X LEFT Markers Knob X MRKR SCALE Markers Knob	Displays the x-marker menu. Rotate the knob to position the x-marker at the *extreme left; time = 0.* Turn on the X "band" markers. Now we're going to expand the time scale to make it easier to see detail. Rotate the knob to position the x-marker on *the right side of the ringing,* where the trace has become flat. This step expands the range between the markers to fill the display. Rotate the knob until the *Δx-marker is positioned at the point where the step response stays within the y-band markers.* The settling time value may now be read directly from the Δx-marker readout. See figure 4-20.

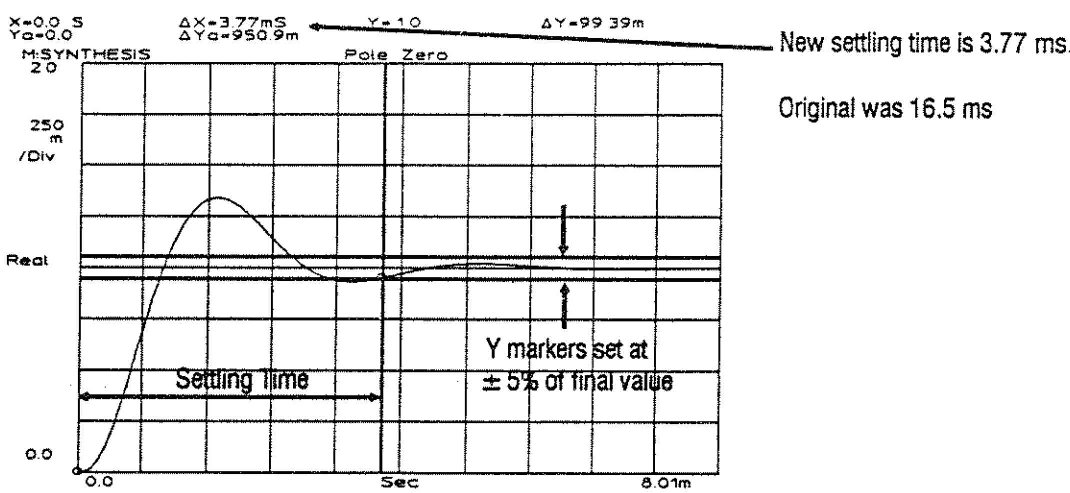


Figure 4-20. Compensated System's Settling Time

Find Rise Time

X OFF **Y OFF**	Turns off both sets of markers. Turning the markers off and on again is a good way to turn off the "band" markers.
Y Markers Knob HOLD Y LOWER Markers Knob	Turn on the y-marker again . Rotate the markers knob to position the *y-marker line at 100 mV.* Turns on the y-band or Δ (delta) marker. Rotate the markers knob until the *ΔY setting is 800 mV.* The y-markers are now positioned at the 10% and 90% levels of the step function.
X Markers Knob HOLD X LEFT Markers Knob	Turn the x-marker back on. Rotate the knob to *position the x-marker* at the 10% value of the rise time (*where the lower y-marker crosses the trace*). Turn on the ΔX marker. Turn the knob to *position the ΔX at the point where the upper y-marker crosses the trace.* The rise time value may now be read directly from the ΔX marker readout. See figure 4-21.

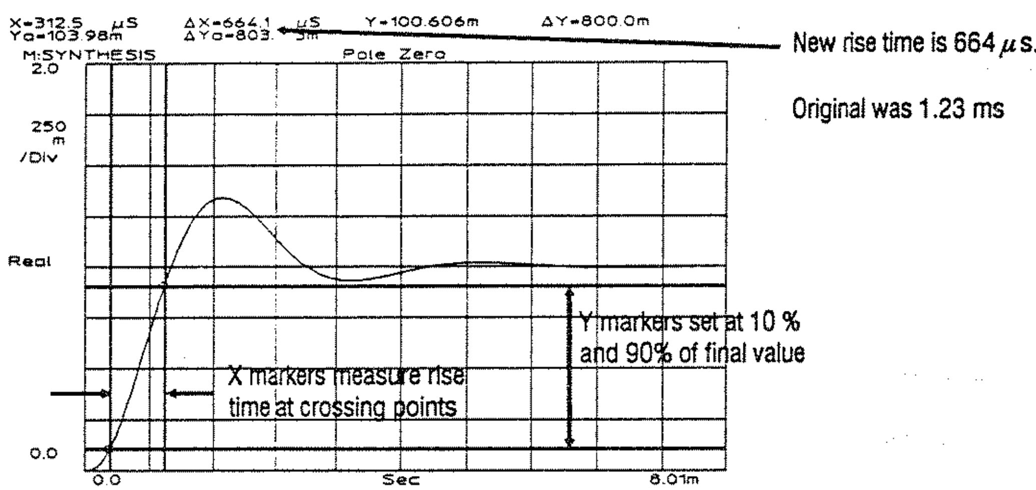


Figure 4-21. Compensated System's Rise Time

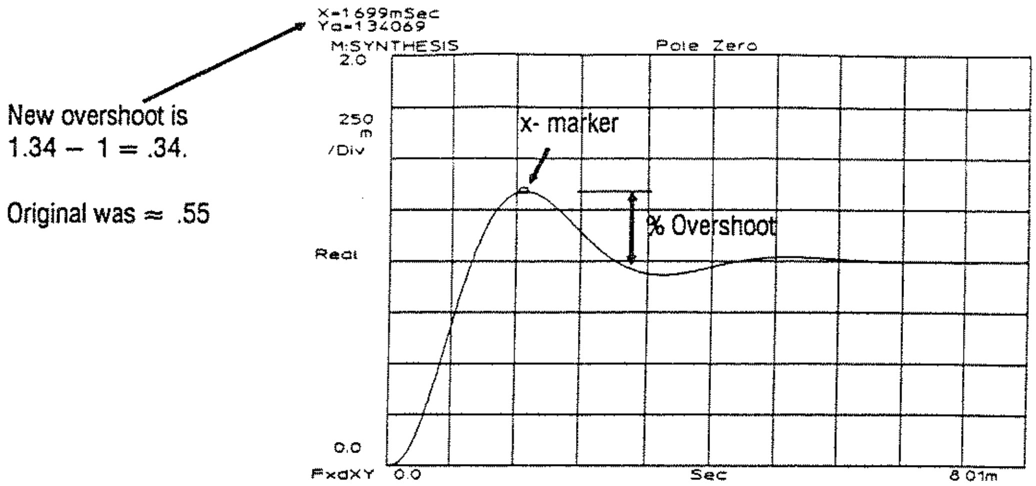


Figure 4-22. Compensated System's % Overshoot

Find % Overshoot

X OFF
Y OFF

Turn off the x-and y-markers.

X

Turns on the x-marker again. It appears at the largest value of the trace, which is the peak of the overshoot. Since the trace has been normalized, the marker value readout, minus one — is the percent overshoot. From a visual inspection of figure 4-4, you can see that the overshoot is about 55 percent.

Step Response Summary

Adding a phase-lead compensator has several effects:

- Improved rise time
- Reduced overshoot
- Reduced settling time

Transform the Design to the Z-Domain

An application may require replacing the analog compensator with a digital filter. Digital circuits are more reliable and are less susceptible to electronic noise and electromagnetic radiation. Systems designers that are more familiar with the s domain may wish to design in the s domain and transform the design results to the z domain. The HP 3563A offers four transformation methods that convert between the s domain and the z domain:

- Bilinear transformation
- Step-invariant transformation
- Impulse-invariant transformation
- Synthesize the trace in one domain and curve fit it in the other domain

Each method generates a different approximation of a continuous system digital filter. This section briefly examines these and chooses one for our task. To do this, we will transform the compensator design using these four methods and synthesize them in the z domain.

First, we'll resynthesize the compensator response and store it. Then we can compare it with results of the transformations.

Configure the Analyzer

MEAS MODE SWEPT SINE	Changes back to the swept-sine measurement mode. (You could also do this in linear resolution mode and select a log axis under Coordinates hardkey. The computational time is longer for the swept sine because the data is logarithmic. In linear resolution, the data is linear; choosing the log axis displays the linear data on a log scale.)
FREQ START FREQ 2 Hz	Checks the starting frequency to ensure synthesis in the correct part of the spectrum.
B COORD PHASE	Sets Trace B to display phase data.
A&B	Selects both traces to receive the results of the synthesis.

Synthesize, then Save, the Original (S-Domain) Compensator Trace

SYNTH POLE ZERO EDIT POLE# **Down Arrow** DELETE VALUE DELETE VALUE DELETE VALUE SYNTH FCTN GAIN FACTOR?	We need to remove the poles of the measured response so that only the compensator's pole and zero remain in the synthesis table.
5.85 ENTER RETURN RETURN	Changes the gain factor back to the compensator design value of 5.85.
CREATE TRACE S DOMAIN	Creates the trace in the s-domain.
A&B **SCALE** Y AUTO SCALE	The display now appears as shown in figure 4-23.
A **SAVE RECALL** SAVE DATA# 1 ENTER	Stores the data in storage register 1.

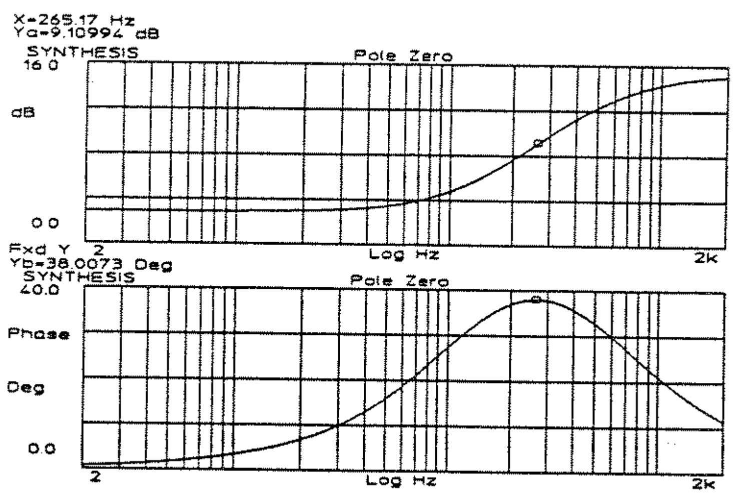


Figure 4-23. Synthesized Compensator Response
This is the "reference" data for following comparisons.

Bilinear Transformation

B **SYNTH** DOMAIN S Z CONVRT TO Z SAMPLE FREQ 15.625 kHz BI-LINEAR (wait ≈ 5 sec) CREATE TRACE Z DOMAIN	These key presses generate z-domain synthesis values used to create the trace in the z domain. The DOMAIN S Z softkey toggles between s and z domains each time you press it; the "S" should be bold and underlined in the display menu. Before the conversion, select the sampling frequency F_s = 15.625 kHz. This is the rate at which the digital system samples data. This value was picked as part of the design of the digital filter. Generally, a sample rate is selected by multiplying the highest frequency of interest by 10. Yields a z-domain synthesis table with a pole at 803.041m, a zero at 949.361m, and a gain factor of 5.41.

Now synthesize a trace from the transform data and compare to reference.

A **COORD** PHASE **FRONT BACK**	In figure 4-24 we now have the phase of the original s-domain trace in Trace A, overlaid with the phase of the trace that was synthesized in the z domain from data that was converted to the z domain with the bilinear transform. Notice the slight variation in the phase at the highest frequency.
A&B **COORD** MAG (dB)	Figure 4-25 now shows the magnitude of the original s-domain trace in Trace A overlaid, with the magnitude of the data that was converted to the z domain with the bilinear transform.

Discussion

This method seems to provide a very good match for the original design. Looking at the marker readout in the figures below we can see that at 2 kHz, the phase difference is about 0.6° and the magnitude difference is insignificant.

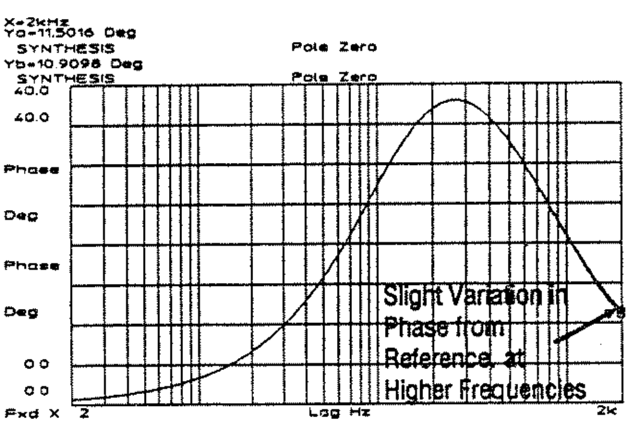


Figure 4-24. Bilinear Transform; Phase

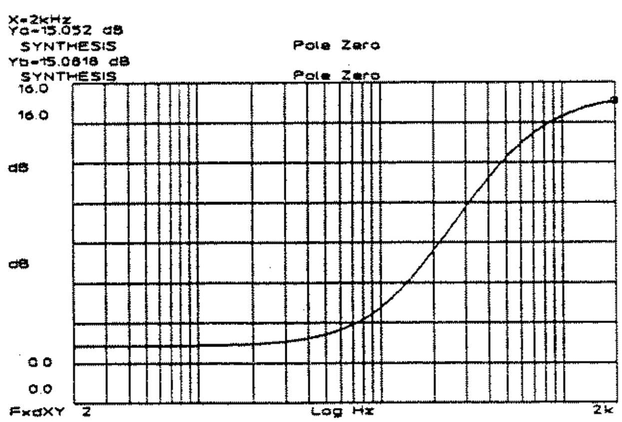


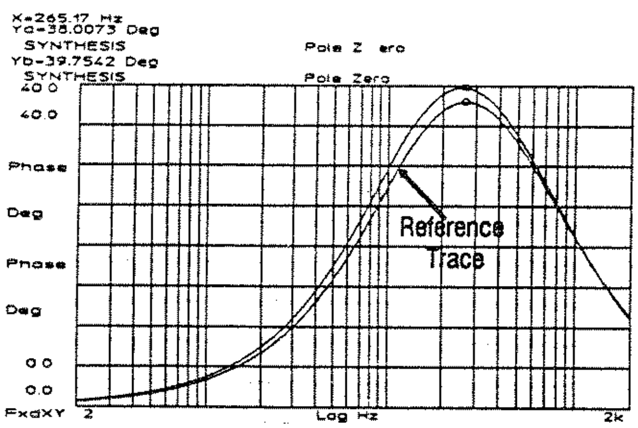
Figure 4-25. Bilinear Transform; Magnitude

Step-Invariant Transformation

B **SYNTH** DOMAIN S Z CONVRT TO Z	These key presses generate pole and zero data in the z domain, and then synthesize the trace so we can compare the results against the reference. See discussion under the bilinear transform example.
SAMPLE FREQ 15.625 kHz	Make sure you enter the correct sample frequency value.
STEP INVRNC CREATE TRACE Z DOMAIN (wait)	This key press performs the transformation. This yields a z-domain synthesis table with a pole at 803.744m, a zero at 953.329m, and a gain factor of 5.85. Now we synthesize the response to compare with the reference.
A&B **COORD** PHASE	Figure 4-26 shows the phase of the original s-domain trace in Trace A, overlaid with the phase of the trace that was synthesized in the z domain from data that was converted to the z domain with the step-invariant transform.
A&B **COORD** MAG (dB)	Figure 4-27 shows the magnitude differences between the original design and the step-invariant transform data synthesized into a trace.

Discussion

This method doesn't do as well as the bilinear transform in terms of FRF match. The phase difference at 265 Hz is 1.7°. The largest magnitude error is 0.72 dB at 2 kHz (the highest measured frequency).



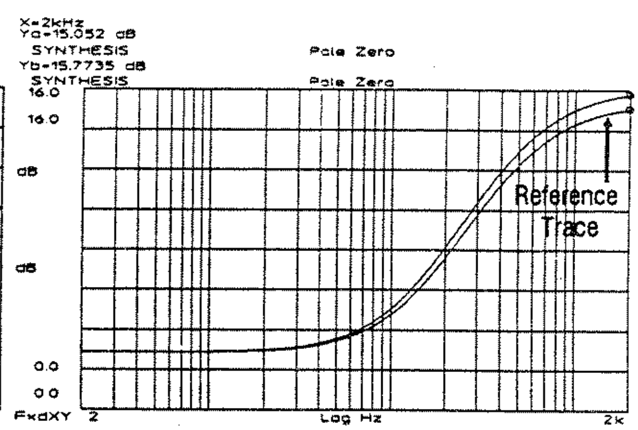


Figure 4-26. Step-Invariant Transform; Phase   **Figure 4-27. Step-Invariant Transform; Magnitude**

Impulse-Invarient Transformation

B
SYNTH
DOMAIN S Z
CONVRT TO Z
SAMPLE FREQ
15.625 kHz
IMPULS INVRNC
(wait ≈ 5 sec)

These key presses generate z-domain synthesis values that are used to create the trace in the z domain.

CREATE TRACE
Z DOMAIN

Yields a z-domain synthesis table in pole-residue form containing a pole at 803.744m with residue – 15.22k and a pole at 0 with a residue of 91.4k.

Now synthesize a trace from the transform data and compare to reference.

A&B
COORD
PHASE
FRONT BACK

In figure 4-28 we now have the phase of the original s-domain trace in Trace A overlaid with the phase of the trace that was synthesized in the z domain from data that was converted to the z domain with the impulse-invarient transform.

A&B
COORD
MAG (dB)

Figure 4-29 now shows the magnitude of the original s-domain trace in Trace A overlaid with the magnitude of the data that was converted to the z domain with the impulse-invarient transform.

Discussion

This transform varies significantly from the original data. This is due to the fact that the original data is not bandlimited. As discussed in the transform summary following these exercises, the impulse-invarient transform introduces distortion due to aliasing and, therefore, should not be used on data that is not bandlimited.

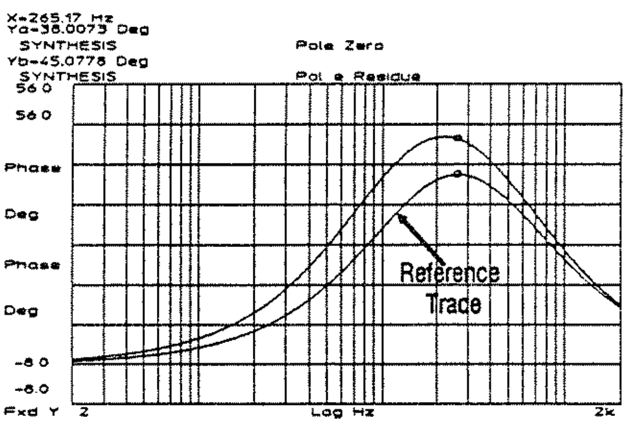


Figure 4-28. Impulse-Invariant Transform; Phase

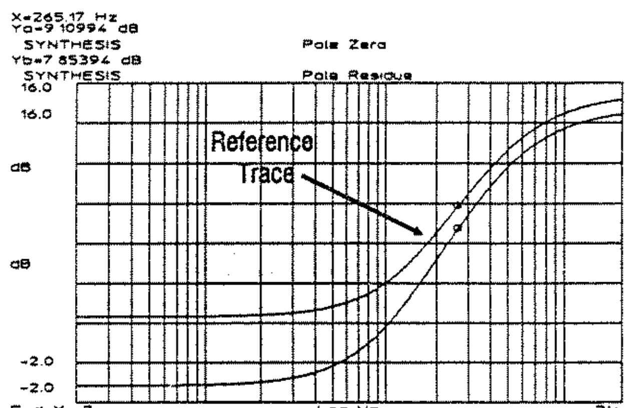


Figure 4-29. Impulse-Invariant Transform; Magnitude

S-to-Z Transformation Using the Z-Domain Curve Fitter

Where the transform functions yield a fixed number of poles and zeros in the destination domain as appeared in the source domain, the curve fitter allows more flexibility in the number of elements used to yield a matching design. See the "Transform Summary" following this exercise.

Assuming that the s-domain compensator design values (pole, zero and gain factor) are still in the s-domain synthesis table, the following steps synthesize magnitude and phase into the A and B traces, and then use the z-domain curve fitter to transform the data to the z domain.

A **COORD** MAG (dB)	Displays magnitude in Trace A (already contains the s-domain compensator response).
B PHASE	Displays phase in Trace B (currently contains the last transform's synthesized response).
SAVE RECALL RECALL DATA#? 1 ENTER	Recalls the s-domain compensator response into Trace B. We now have the compensator response in the A and B traces.
XOFF	Be sure the x-marker is off. When on, the curve fitter uses only data near the marker position — here we want it to fit the entire response spectrum.
CURVE FIT	Configures the analyzer for curve fitting the s-domain response.
DOMAIN S **Z**	Select the z domain.
A&B TRACES	The data to curve fit is in A and B traces (not the last measurement).
NUMBER POLES? 1 ENTER	Limits the number of poles to 1.
NUMBER ZEROS? 1 ENTER	Limits the number of zeros to 1.
EDIT TABLE TABLE FCTNS SAMPLE FREQ 15.625 kHz RETURN RETURN	Enters the sample frequency specified by the digital filter design.
CREATE FIT START FIT (wait; ≈ 1 min)	Begins the fit. After the fit is complete, the response in Trace B is the best fit.
FRONT BACK	Overlays the traces.
A&B	Activates both traces.
X	Turn on the x-markers. See results in figure 4-31.
COORD PHASE	Changes the coordinates of both traces to display phase. See the results in figure 4-30.

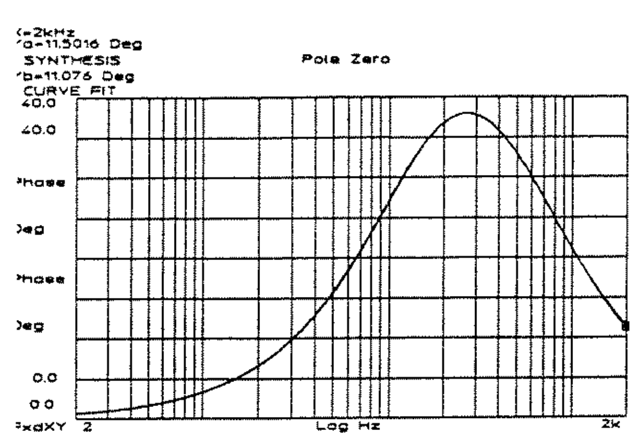


Figure 4-30. Curve Fit "Transform"; Phase

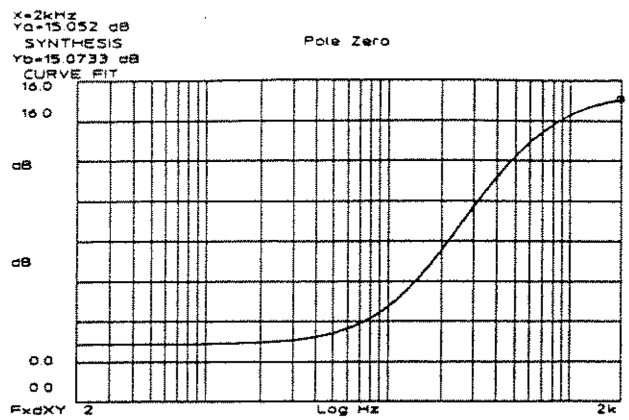


Figure 4-31. Curve Fit "Transform"; Magnitude

Now we'll take a look at the values in the curve fit table.

CURVE FIT
EDIT TABLE — These key presses display the curve fit table shown in figure 4-32

Discussion

This method also seems to give very good results. The magnitude and phase differences, as well as the pole and zero values, are very nearly identical to the bilinear transform results. Before recording these results, we curve fit once with the number of poles and zeros set to five each. The results included one unstable pole — magnitude larger than 1— outside the unit circle. Note that these results are approximately the same as we found with the bilinear transform.

Z Curve Fit

	POLES 1	ZEROS 1
1	800.181m	948.564m
	Time delay=0.0 S Gain=5.4	Sampl=15.6k +Zpwr

Figure 4-32. Curve Fit "Transform" Results Table

Transform Summary

The goal of transforming between the s and z domains is that the imaginary axis of the s-plane maps into the unit circle of the z-plane. Also, transforming a stable filter in one domain should result in a stable filter in the other domain — stable poles in the s-plane (in the left half of the plane) transform to stable poles in the z domain (inside the unit circle).

Impulse-Invariant Transform

The basis for this method is to choose a unit-sample response in the z domain filter that approximates the impulse response of the analog filter. The impulse-invariant transform preserves the relationship between analog and digital frequency response, but introduces distortion due to aliasing. Thus, this method is only useful for bandlimited filters.

Step-Invariant Transform

This method is similar to the impulse-invariant approach. However, instead of approximating the impulse response of the analog filter, the step-invariant transform preserves the step response characteristics of the original filter.

Bilinear Transform

The bilinear transform yields stable digital filters from stable analog filters (and vice versa). It also avoids the problem of aliasing characteristic of the impulse-invariant transform. It does, however, introduce distortion in the frequency axis. The use of the bilinear transform is useful only when this distortion can be tolerated or compensated.

Curve Fitting Between the S and Z Domains

None of the transforms listed above are completely accurate. Conversion from the s domain to the z domain can often be done better by fitting the measured data directly using the z-domain curve fitter. This approach generally gives a good fit over the entire frequency range of interest. Also, the measured data can be manipulated (with trace math) to remove errors or to compensate for other components in the system — such as time delays or zero-order hold effects — before the fit is calculated. In addition, you can use the weighting function to emphasize the more-important regions of the response and to ignore or de-emphasize less important regions.

The main disadvantages of this method are the potential for unstable poles (outside the unit circle), and the possibility of obtaining a non-minimum phase-transfer function. Also, there is no explicit control over the resultant filter time-response.

For more information, refer to appendix D, *Z-Domain Curve Fitting in the HP 3563A Analyzer.*

Testing and Refining the Digital Filter Design

Sampled System Effects

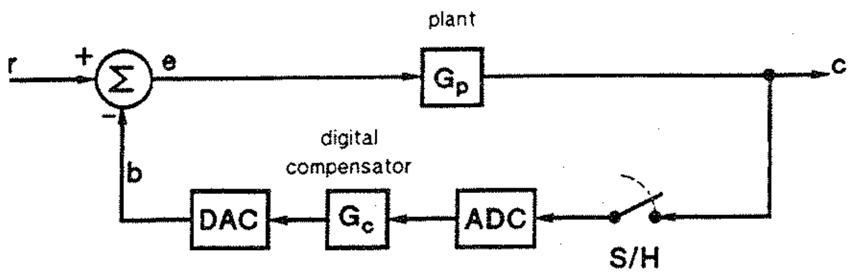


Figure 4-33. Digital Compensator Block Diagram

Computational Delay

Our digital filter consists of a microprocessor that reads in sampled data, processes it, and outputs data. The time it takes to process the data is called *computational delay*. In the frequency domain, this appears as a downward-sloping phase ramp. Increasing delay causes more rapid phase rolloff.

Zero-Order Hold

When the digital filter is placed in our mostly-analog system, it is followed by a digital-to-analog converter that has a filtering effect called *zero-order hold* (ZOH). We must consider its effects on the signals coming from the digital filter, since it is part of the compensator.

The filter effect of the zero-order hold varies depending on the sampling rate chosen. You can see the different phase responses of two sampling rates in figure 4-34. Figure 4-35 shows the familiar $\frac{\sin x}{x}$ magnitude responses of a ZOH where F_s = 15.625 kHz and 31.25 kHz and the span is 100 kHz. Note that raising the sample rate (F_s) rate reduces the effects of the ZOH over the 0 Hz–2 kHz span (markers are set at our 2 kHz span limit). Increasing F_s also raises the cost of the design. A general rule used to estimate the required sampling frequency is 10 × the highest frequency of interest.

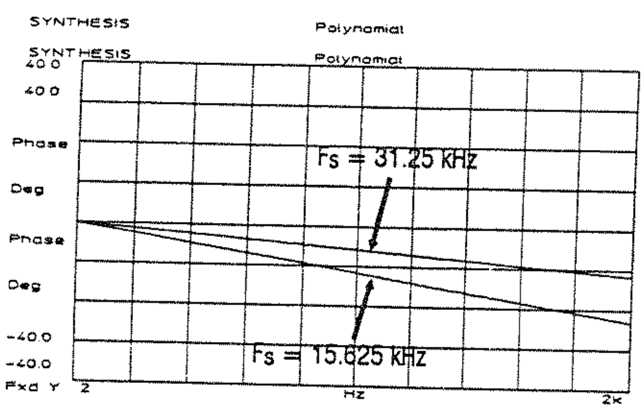


Figure 4-34. ZOH Phase at Two Sample Rates
2 Hz-to-2 kHz

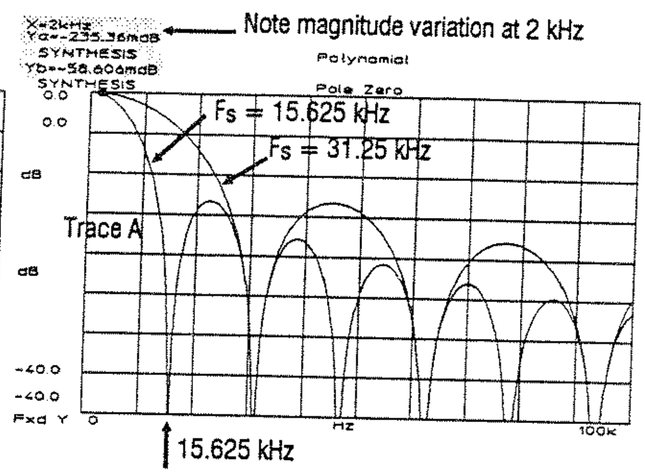


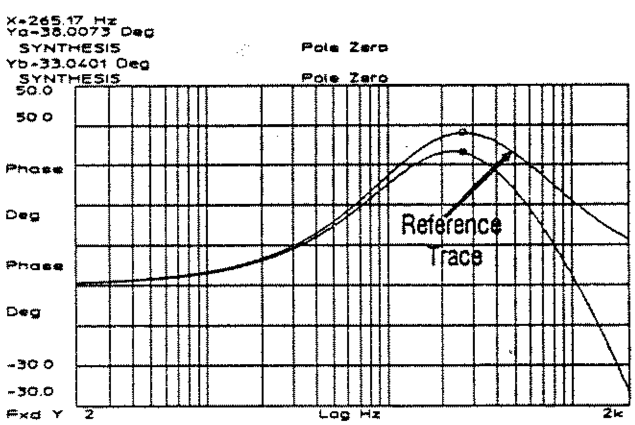
Figure 4-35. ZOH Magnitude Varies with F_s
0 Hz-to-100 kHz

Synthesize the Transform Data

When we compared the transformed data to the reference, we synthesized it to get a response trace. In the case of the curve fit "transform," the curve fitter provided us with a trace — we didn't have to synthesize one. None of the comparisons took the computational delay and ZOH effects into consideration. We will synthesize the curve fit data, with the delay and ZOH effects added, to get a realistic simulation of the response of the digital filter section.

Assuming our earlier curve fit data is still in the curve fit table:

CURVE FIT DOMAIN S **Z** FIT FCTN FIT→SYNTH	Moves the curve fit data to the synthesis table.
B **SYNTH** POLE ZERO SYNTH FCTN TIME DELAY 12 uSec	Activates B so that the following synthesized trace is displayed there. Enters a time delay of 12 μs. We are estimating that the digital filter design requires roughly 12 μs for a 1-pole 1-zero solution. If more accurate data becomes available on a later design cycle, we'll use it instead.
SAMPLE FREQ 15.625 kHz RETURN RETURN	Sets the sample frequency to 15.625 kHz.
CREATE TRACE 0 HOLD <u>ON</u> OFF Z DOMAIN	Creates the trace with zero-order hold turned on. This synthesizes a trace with the ZOH effects included.
FRONT BACK **A&B** **COORD** PHASE	Figure 4-36 compares the phase of the reference trace with that of the trace derived by transforming the reference data to the z-domain, and then synthesizing it with the effects of ZOH and delay added. Figure 4-37 compares the magnitude of the two traces. You can see that phase response was affected much more severely than the magnitude.



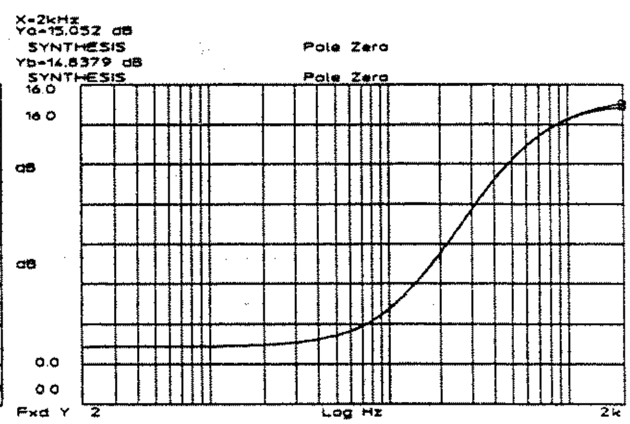


Figure 4-36. ZOH and Delay Effects on Phase

Figure 4-37. ZOH and Delay Effect on Magnitude

Check Results Against the Specifications

Now we will combine the digital filter data with the control system data and check the specifications we are trying to achieve. To do this, we will put the control system poles and gain factor in the s-domain synthesis table and synthesize it with the digital filter data already in the z-domain synthesis table. With the "Z&S" synthesis capability, the data for the two do not have to be in the same table.

UPPER LOWER
A&B

We'll put the combined response in both traces and make Trace A show magnitude and Trace B show phase.

SYNTH
DOMAIN <u>S</u> Z
POLE ZERO

Adds the pole and gain factor values shown in figure 4-12 back into the s-domain synthesis table.

CLEAR TABLE
CLEAR TABLE

Clearing the table takes two key presses. This reduces the chance of accidently deleting valuable data.

EDIT POLE#
ADD VALUE
– 884.303 Hz
– 72.6399Hz
– 15.9307Hz
SYNTH FCTN
GAIN FACTOR
23.7 EXP
6 ENTER
RETURN
RETURN

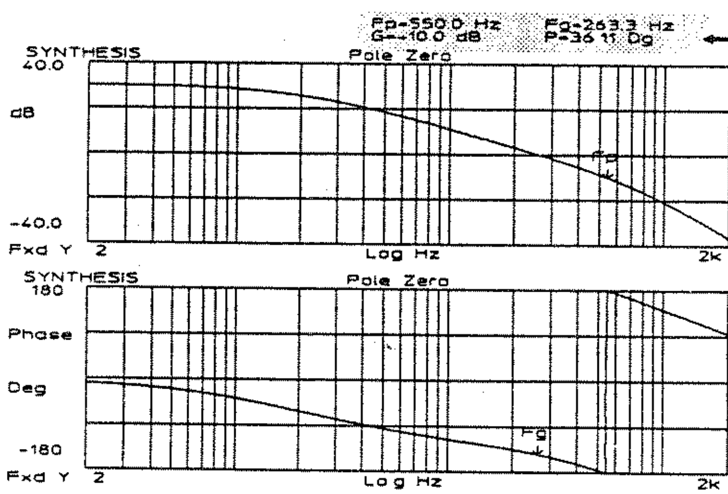
Instead of entering this data again we could have curve fit a measurement; that is how we got the data in the first place.

CREATE TRACE
0 HOLD <u>ON</u> OFF
Z & S DOMAIN

Creates the trace. Other settings (under z-domain, synth fctn) assumed here:
Sample frequency = 15.625 kHz
Time delay = 12 μs

XOFF
SPCL MARKER
MARKER CALC
GAIN & PH MGN

Make sure the x-marker is off. If it's on this limits the range of calculations. Calculates the gain and phase margins. Results shown in figure 4-38.



Special markers gain and phase margins:

Crossover = 263 Hz
Gain margin = – 10.0 dB
Phase margin = 36.1°

Figure 4-38. Checking Margin and Crossover Specs

Parameter	Goal Specification	Results
Gain crossover	265 Hz ± 20 Hz	263 Hz
Gain margin	– 12 dB ± 2 dB	– 10.0
Phase margin	40° ± 5°	36.1°

Comparing the results in figure 4-38 (see special marker settings) with the original specifications we see that our present solution is very marginal. The gain crossover is very close to the goal of 265 Hz. The phase margin is within tolerance, but only by 1.1°.

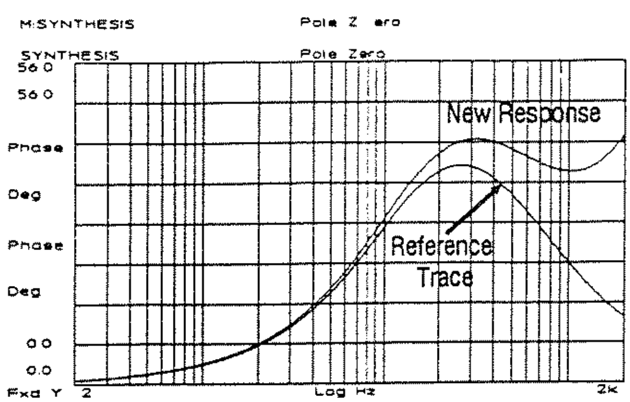
Conclusion: Replacing an analog compensator with a digital solution must take into account the effects of sampling (ZOH) and computational delay. These effects can be reduced by sampling at faster rates and using a faster microprocessor for the digital filter — but these solutions increase the cost of the design.

Next we will try another method of compensating for the side effects of using a digital filter that does not increase the order of the system (for example, trying try to find another 1-pole, 1-zero solution).

Compensating for Sampling Effects

One way to compensate for the phase and magnitude rolloff caused by ZOH and computational delay is to synthesize these effects to create a trace, and then divide our reference trace by it. This generates a "new response" that will cancel out the digital filter side effects. We'll put the reference response data back in the s-domain synthesis table and create its response in the s domain. Then we will synthesize the side effects response in the z domain.

A SYNTH DOMAIN S Z POLE ZERO CLEAR TABLE CLEAR TABLE EDIT POLE# ADD VALUE – 543.3 Hz EDIT ZERO# ADD VALUE – 129.2 Hz SYNTH FCTN GAIN FACTOR 5.85 ENTER RETURN RETURN CREATE TRACE S DOMAIN	We're going to put the reference response in Trace A.
B SYNTH DOMAIN S Z POLYNOMIAL EDIT NUMER# ADD VALUE 1 ENTER SYNTH FCTN TIME DELAY 12 uSec SAMPLE FREQ 15.625 kHz RETURN RETURN CREATE TRACE 0 HOLD ON OFF Z DOMAIN	These key presses create the side effects response in trace B. We've already had a look at the ZOH response in figures 4-34 and 4-35. The delay effects are pure phase ramp. This method creates a "wire filter" which has a perfectly flat response. When we synthesize it with ZOH on, and a non-zero time delay value, we get a response due to the side effects of computational delay and ZOH.
A&B COORD PHASE	Since phase seems to be the problem area, we will monitor the phase response.



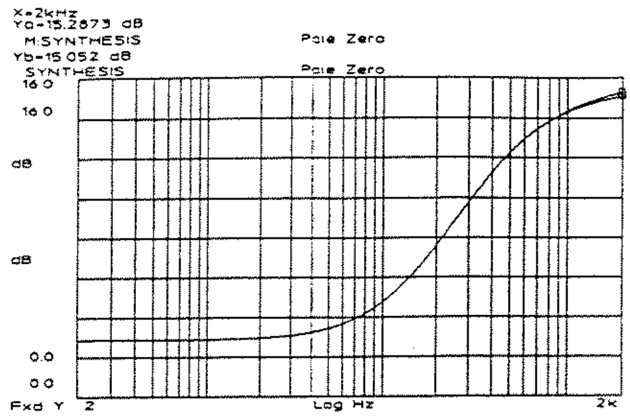


Figure 4-39. A New Response Phase

Figure 4-40. A New Response Magnitude

A **MATH** DIVIDE TRACE B	This divides Trace A (the original reference) by trace B. When this response is compared with the original, we see in figures 4-39 and 4-40 that the phase of the new response is as advanced as the earlier comparison was delayed. A digital filter with this response will cancel the sampled system side effects.
SAVE RECALL SAVE DATA#? 2 ENTER	Saves the new response in a data register. This will save time when we need this trace later.
X X VALUE? 100 Hz HOLD X LEFT	Turns on the x-markers; we will use x-band markers to define the area over where we want the curve fit to operate — from 100 Hz to 400 Hz. This overrides the weighting function.
Marker Knob	*Rotate the knob until the ΔX marker is ≈ 300 Hz.*
CURVE FIT	Curve fit, constraining the number of poles and zeros to 1 each.
DOMAIN S **Z**	This makes sure we are in the z domain.
A&B TRACES	This curve fits the data in Trace A. (This uses Trace B if it contains coherence data).
NUMBER POLES 1 ENTER	Sets number of poles to 1.
NUMBER ZEROS 1 ENTER	Sets number of zeros to 1.
EDIT TABLE TABLE FCTNS SAMPLE FREQ? 15.625 kHz RETURN RETURN	Set the sample frequency to 15.625 kHz.
CURVE FIT START FIT	Starts the curve fitter.

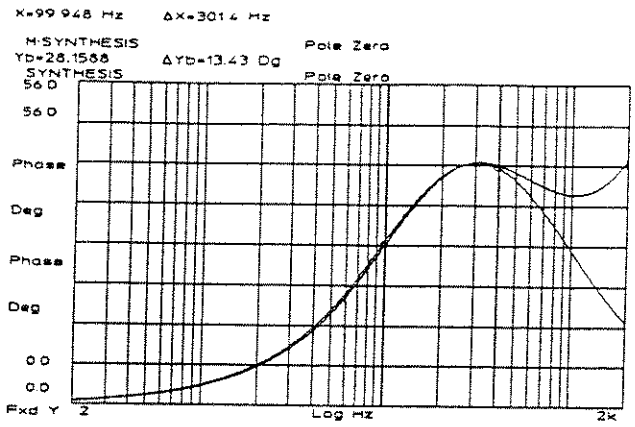


Figure 4-41. 1-Pole, 1-Zero Curve Fit Phase

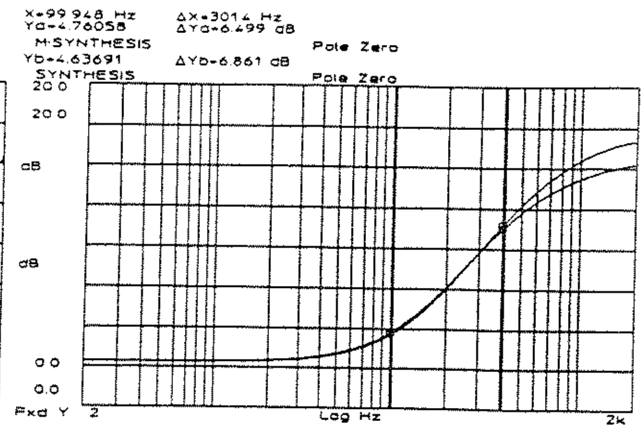


Figure 4-42. 1-Pole, 1-Zero Curve Fit Magnitude

CURVE FIT
FIT FCTNS
FIT→ SYNTH
B
SYNTH
CREATE TRACE
0 HOLD ON OFF
Z DOMAIN

Since the curve fit was limited by the x-band markers, the only part of the new trace generated by the curve fitter was between the markers. To see the response of the results over the entire span, we will synthesize it in Trace B. First, copy the data to the synthesis table. Then synthesize the trace and compare with the new response we are trying to match.

The comparison is shown in figures 4-41 and 4-42. Note the accuracy of the fit between the vertical marker lines.

Next we will synthesize this result with the original system response to check specifications.

A — Selects Trace A to receive synthesis results.

SYNTH
DOMAIN S Z
POLE ZERO
CLEAR TABLE
CLEAR TABLE
EDIT POLE#
ADD VALUE?
− 884.303 Hz
− 72.6399 Hz
− 15.9307 Hz
SYNTH FCTN
GAIN FACTOR?
2.37 EXP
6 ENTER
RETURN
RETURN

Enters the system's poles and gain factor in the s-domain synthesis table. See curve fit table in figure 4-12.

CREATE TRACE
S DOMAIN

Synthesizes the s-domain response in Trace A for comparison.

DOMAIN S **Z** POLE ZERO SYNTH FCTN TIME DELAY 12 uSec SAMPLE FREQ 15.625 kHz RETURN RETURN	Next, we check the z-domain parameters.
B CREATE TRACE	Selects Trace B to display the combination synthesis.
0 HOLD <u>ON</u> OFF Z & S DOMAIN	Synthesizes the two tables into one response (ZOH on and delay = 12μs).
X OFF	Make sure the x-marker is off.
SPCL MARKER MARKER CALC GAIN & PH MGN	Checks the gain-crossover and margins against specifications. See the results in figure 4-43.

Parameter	Goal Specification	Results
Gain crossover	265 Hz ± 20 Hz	265.0 Hz
Gain margin	− 12 dB ± 2 dB	− 11.1 dB
Phase margin	40° ± 5°	40.1°

Conclusion: This design appears to meet the goal specifications. The only concern is the accuracy of the computational delay. The next step is to implement the digital filter (write the program) and test the results. When the filter is working, we can measure the system response and the actual computational delay. If the implementation does not meet specifications, we should consider the difference in assumed and measured computational delay, and run through the latter stages of this design loop once again.

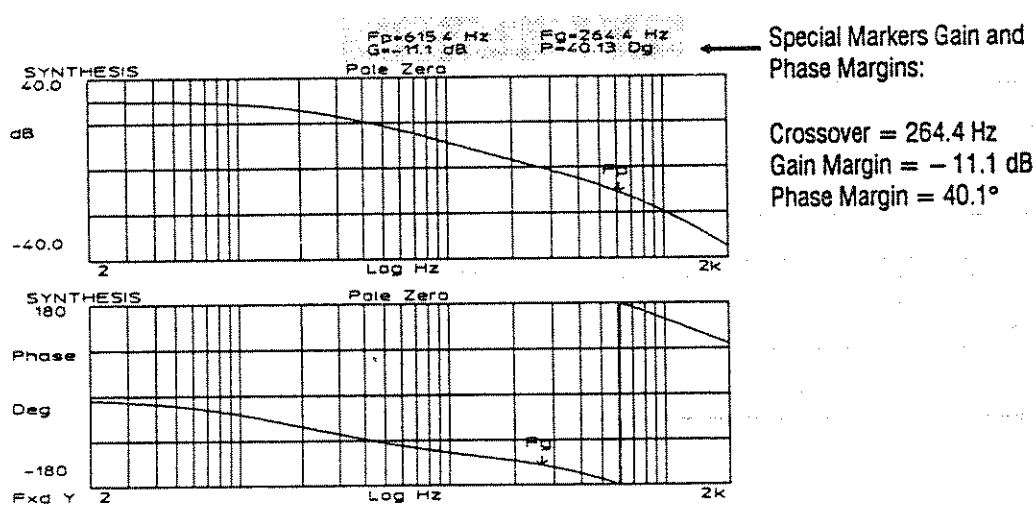


Figure 4-43. Synthesized Trace of Final Design

Final Results

The digital filter was constructed, programmed, and inserted in the feedback loop of the control system as shown in figure 3-11. The measurement results appear below.

Figure 4-44 shows the measurement of the gain crossover and gain and phase margins. The special marker values show that the implementation closely matches the design results. This measurement was taken as described earlier, with the exception that the source level was reduced to 100 mV, to compensate for the gain added to the feedback loop.

Figures 4-45 and 4-46 show the rise time and settling time of the system . These measurement were taken as described earlier, except for the source level, which was 200 mV instead of 1 V. Note that there is no overshoot. The marker values show a rise time of 265 μs and a settling time of $\approx$ 2.2 ms

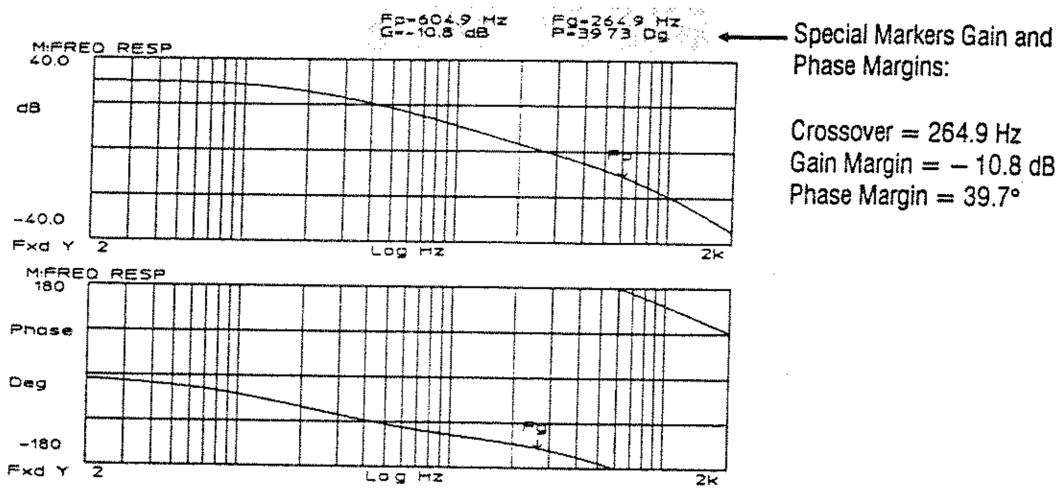


Figure 4-44. Specifications Check of Digital Filter

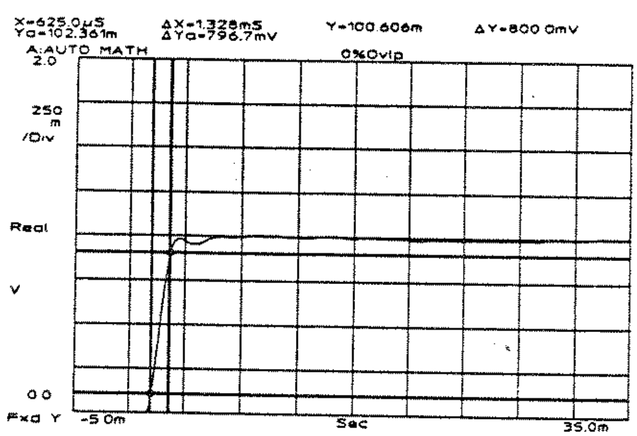


Figure 4-45. Rise Time Measurement

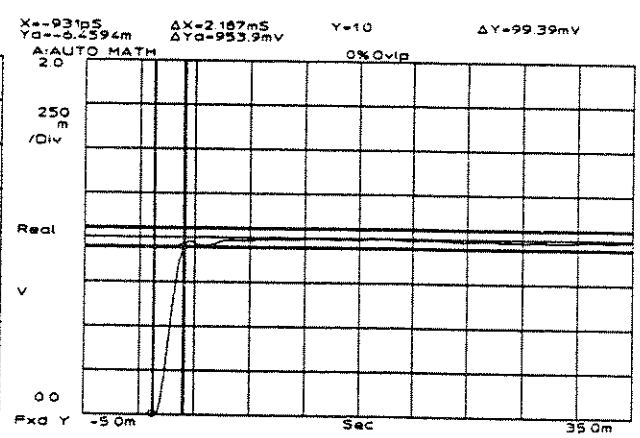


Figure 4-46. Settling Time Measurement

Index

K

L

M

N

O

P

Q

R

S

Hewlett-Packard Sales and Service Offices

To obtain Servicing information or to order replacement parts, contact the nearest Hewlett-Packard Sales and Service Office listed in HP Catalog, or contact the nearest regional office listed below:

In the United States

California
P.O. Box 4230
1421 South Manhattan Avenue
Fullerton 92631

Georgia
P.O. Box 105005
2000 South Park Place
Atlanta 30339

Illinois
5201 Tollview Drive
Rolling Meadows 60008

New Jersey
W. 120 Century Road
Paramus 07652

In Canada
Hewlett-Packard (Canada) Ltd.
17500 South Service Road
Trans-Canada Highway
Kirkland, Quebec H9J 2M5

In France
Hewlett-Packard France
F-91947 Les Ulis Cedex
Orsay

In German Federal Republic
Hewlett-Packard GmbH
Vertriebszentrale Frankfurt
Berner Strasse 117
Postfach 560 140
D-6000 Frankfurt 56

In Great Britain
Hewlett-Packard Ltd.
King Street Lane
Winnersh, Wokingham
Berkshire RG11 5AR

In Other European Countries
Switzerland
Hewlett-Packard (Schweiz) AG
7, rue du Bois-du-Lan
Case Postale 365
CH-1217 Meyrin

In All Other Locations
Hewlett-Packard Inter-Americas
3155 Porter Drive
Palo Alto, California 94304

EDN

Designer's Guide to:
Linear control-system theory—Part 1

Apply control-system theory to analyze closed-loop systems

Reprinted from **EDN** April, 1986

Apply control-system theory to analyze closed-loop systems

Dynamic signal analyzers (DSAs), which analyze signals in both the time and frequency domains, aid you in taking control-system measurements and in performing other steps in system development, such as analysis, modeling, and design. Part 1 of this 3-part article presents an overview of classical linear control theory. Part 2 will examine both old and new methods of taking the actual measurements, and part 3 will explore the expanded role of DSAs in the control-system design process.

Steve Asbjornsen and Owen Brown,
Hewlett-Packard Co

Control-system development is largely a study of the operating characteristics of devices and the ways in which the devices interact when combined in a system. You can apply linear control-system theory to analyze any system that uses deliberate guidance or manipulation to achieve a specific value for some variable, whether it's electrical, mechanical, or biological. These systems can range from configurations as simple as Watt's flyball governor to networks whose analysis requires calculations that only a computer can handle adequately. Examples of familiar control systems include motor-speed controls, pacemakers, voltage regulators, switching power supplies, and phase-locked loops.

The most fundamental distinction in control theory is the classification of systems. Without exception, systems fall into two categories: open-loop and closed-loop systems. Open-loop control systems (**Fig 1a**) are ones that use a controller that has no feedback from the output. These systems can't take corrective action to alleviate undesired changes in the output. Closed-loop systems (**Fig 1b**) are ones whose output is fed back and compared with the input in such a way as to maintain the desired output. This series of articles will consider only closed-loop systems.

You can represent any closed-loop system with the standardized diagram shown in **Fig 1b**. In the diagram, the output C (the directly controlled variable) feeds back through a functional block with transfer function H and is compared with reference signal R at a summing junction. The difference between R and t[illegible] feedback signal (B) is the error, or actuating signal (E[illegible] The reaction of the components represented by G [illegible] response to error signal E maintains the output at the desired level.

If controlled variable C is fed back to the summing

Easy to implement in a test system, the step forcing function reveals a wealth of information about a system's speed, stability, and settling time.

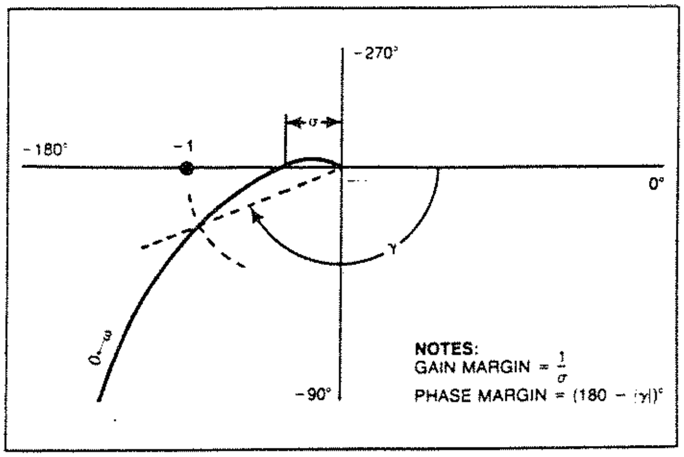


Fig 8—Measuring gain and phase margin with a Nyquist diagram is easy. The gain margin is the additional gain needed to achieve unity gain at 180° phase shift, and the phase margin is the additional phase shift needed to achieve 180° at the unity-gain frequency.

achieve unity gain at the frequency where the phase of the open-loop frequency response equals 180° as it crosses the negative real axis (**Fig 8**). Simply stated, the gain margin is the additional gain required to place a closed-loop pole on the jω axis (GH(jω)=−1) and produce instability.

Gain margin alone is not sufficient for you to determine how close GH(jω) is to the −1 point and, therefore, how close a closed-loop pole is to the jω axis, as **Fig 8** shows. You also need to use phase margin, which is the additional phase shift required to achieve 180° of phase shift at the highest frequency at which the open-loop gain is unity. Typical target values are a gain margin of not less than two (6 dB) and a phase margin of not less than 30°.

The Nyquist diagram is a powerful tool for analyzing all types of systems, but it is not without shortcomings. Its primary limitation is that it lacks a convenient graphical technique that you can use to predict how changes in the system will affect the open-loop frequency response. Any time the system is changed, you must remeasure the response or recalculate it from your system's model. Bode developed a diagram that alleviates this problem.

The Bode diagram (**Fig 9**) is also a plot of the open-loop frequency response, except that Bode treated gain and phase separately, plotting each as a function of frequency (ω). This technique has several advantages for the designer who must evaluate systems while developing them.

By expressing gain in logarithmic units (dB) and phase in degrees, the Bode diagram allows you to

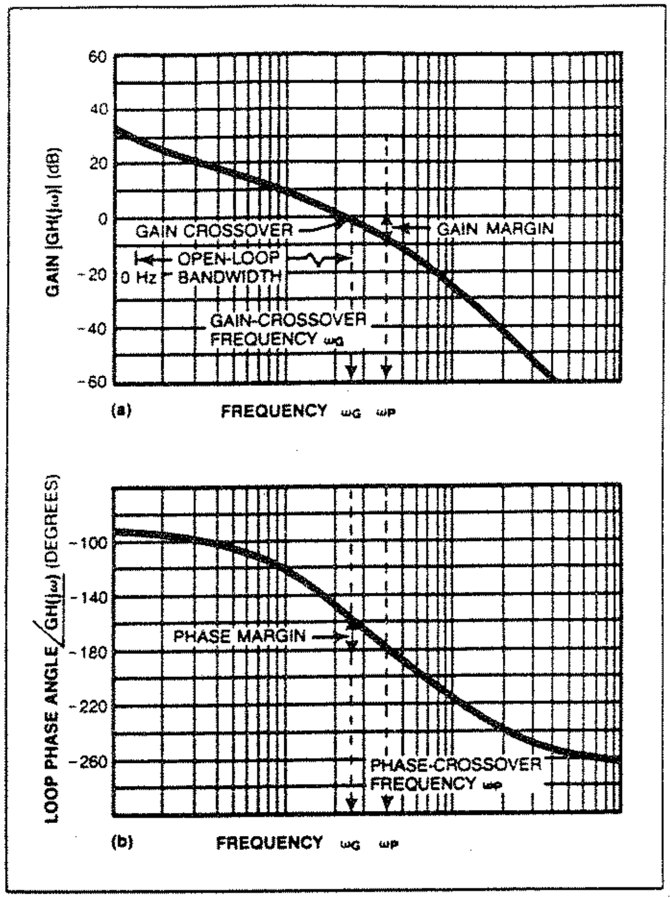


Fig 9—A Bode plot yields gain and phase margin and bandwidth. The open-loop bandwidth is defined here as the frequency at which the system has unity gain.

combine new data with already established measurements through simple addition. The resulting trace is equivalent to the frequency response you would obtain by connecting the actual devices in cascade and physically measuring the composite.

Because of certain techniques that Bode developed, engineers could sketch an approximation of a device's frequency response from its transfer function—ie, they could evaluate GH(jω) over a range of frequencies from the equation for GH(s)—in a Bode diagram without taking the physical measurements. Designers could also perform the opposite function, that of approximating a transfer function from a measured response. For the first time, engineers could link a system's model to measured data without having to resort to extremely laborious calculations.

Bode plots like the ones in **Fig 9** let you measure a system's gain and phase margins. A third performance parameter, the open-loop bandwidth, is also easy to

Mathematical techniques by Laplace and Heaviside, and graphical-analysis methods by Nyquist, Bode, and Nichols aid you in predicting closed-loop system response.

Dynamic signal analyzers

Dynamic signal analyzers (DSAs) are low-frequency analyzers that sample signals applied to their inputs and then use Fourier transforms to analyze the signals in both the time and frequency domains. Useful to engineers who take mechanical, acoustic, and audio-electronics measurements, these analyzers record waveforms and take high-speed, single-channel frequency-spectrum and 2-channel frequency-response measurements.

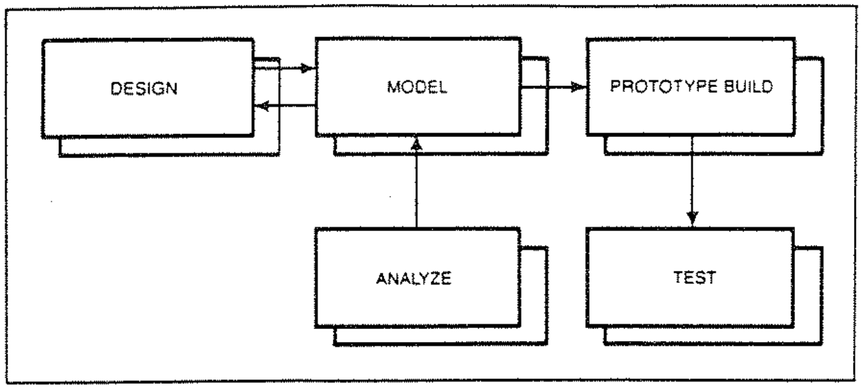


*Fig A—**In a typical product- or system-development cycle,** these five processes all play crucial roles. A dynamic signal analyzer (DSA) expands the traditional role of test instruments from the analyzing and testing functions to include significant contributions to the model and design processes.*

Because they've recently acquired an array of advanced measurement and analysis functions that are directly applicable to control-system measurements, DSAs are replacing frequency-response analyzers in measurement and performance analysis.

However, DSAs are no longer simply measurement instruments; they have new analytical capabilities that, in many respects, equal those of workstations (**Fig A**). These capabilities allow designers who use DSAs to measure a device under test and to model, revise, and perfect the response of the as-yet-unbuilt system.

oped. A magnitude contour (**Fig 7a**) is a locus of points on the Nyquist diagram for which the ratio V_2/V_1 has the same magnitude. When you plot loci for several values of magnitude, the loci form a family of circles. When you plot the open-loop transfer function against these contours, you can identify the gain of the closed-loop transfer function by its intersection with or proximity to a given magnitude contour (**Fig 7b**). You can draw similar loci, called phase contours, for positions of constant phase, and you can use them the same way you use magnitude contours.

Gain, phase margins

The basic Nyquist diagram generated several other graphical measurement aids. Eight years after Nyquist published his techniques, Hendrik Bode wrote a paper in which he observed that systems should be designed to be absolutely stable, one of two possible states for a stable system. An absolutely stable system is one that has some unique value of frequency-independent gain (K_1), below which the system will always be stable, and above which the system will always be unstable.

Alternatively, a conditionally stable system is one with a gain (K_2) above which the system will generally become unstable. In a conditionally stable system, there is also a band of gains greater than K_2 where the system becomes stable again. However, operating in this higher band requires careful control of gain levels; for this reason, Bode and others considered operating in this band undesirable.

In an absolutely stable system, the open-loop frequency response should cross the negative real axis only between 0 and the −1 point. Bode's observations simplified stability evaluations even further by eliminating the need to plot the frequency response's complex conjugate.

Bode also observed that, as the plot of the open-loop frequency response approaches the value of −1, the real part of a closed-loop pole approaches the value of 0. A system with such a response has longer settling times and, in general, less stability. Therefore, Bode established phase-margin and gain-margin parameters to provide a quantifiable measure of the proximity of th open-loop frequency response to this −1 point.

Bode viewed the Nyquist diagram as a polar plot which gain is the distance from the origin and phase is measured as an angle with 0° as the positive real axis. The gain margin is the additional gain required to

Polynomials within polynomials

The terms G(s) and H(s) in the transfer function for a closed-loop system are themselves generally ratios of polynomials in s. You can thus define the terms by using the following expressions:

$$G(s) = \frac{G_n(s)}{G_d(s)}$$

$$H(s) = \frac{H_n(s)}{H_d(s)},$$

where the subscripts n and d indicate the numerator and denominator portions of G(s) and H(s), respectively. If you reformulate the closed-loop transfer function in terms of the numerator and denominator of G(s) and H(s), you obtain the following expression:

$$\frac{C(s)}{R(s)} = \frac{G(s)}{1 + GH(s)} = \frac{\frac{G_n(s)}{G_d(s)}}{1 + \frac{G_n(s)\,H_n(s)}{G_d(s)\,H_d(s)}} = \frac{\frac{G_n(s)\,H_d(s)}{G_d(s)\,H_d(s)}}{\frac{G_d(s)\,H_d(s) + G_n(s)\,H_n(s)}{G_d(s)\,H_d(s)}}.$$

You express the closed-loop transfer function in this manner to illustrate that the term 1+GH(s) itself has poles and zeros, and that the zeros of this term determine the poles of the closed-loop transfer function. Note also that the zeros of the closed-loop transfer function are the roots of the equation $G_n(s)H_d(s)=0$.

If you could measure GH at each value of s you could simply determine when GH equals −1 and record those values of s. Unfortunately, the only values of GH you can physically measure are those values of s for which the real part equals 0—ie, GH(0+jω), which is commonly expressed as GH(jω). All other values of s simply provide an analytical model for understanding how a system's components will affect its time- and frequency-domain responses. Evaluating GH(jω) provides the input/output relationship of the device, typically in the form of gain and phase shift as a function of frequency. GH(jω), therefore, is called a frequency response, and it represents the frequency-domain characteristics of the system.

Nyquist's stability criterion

Using only the information from evaluating GH(jω), Nyquist had to determine whether there were any values of s (having positive real parts) that satisfied the equation GH(s)=−1. Of course, if GH(jω)=−1 at some frequency ω_1, then it's clear that a closed-loop pole exists at $s=(0+j\omega_1)$. But determining whether there were other closed-loop poles having positive real parts was not easy.

Nyquist's contribution lay in his discovery of a technique in which the closed-loop system's measured open-loop frequency response could be used to determine the existence of any closed-loop poles with positive real parts. The mathematical proof behind Nyquist's stability criterion is not intuitively obvious. However, the graphical representation of this discovery, known as a Nyquist diagram, is simple to use, and it was soon adopted by the engineering community.

In a Nyquist diagram, you plot a system's open-loop frequency response on a graph whose coordinates represent real and imaginary components. To derive this trace, you measure device response at a number of different frequencies. To complete the Nyquist diagram, you also plot the complex conjugate of the response on the same graph, typically using dashed lines (**Fig 5a**).

Nyquist's criterion states that if you affix the tail of a

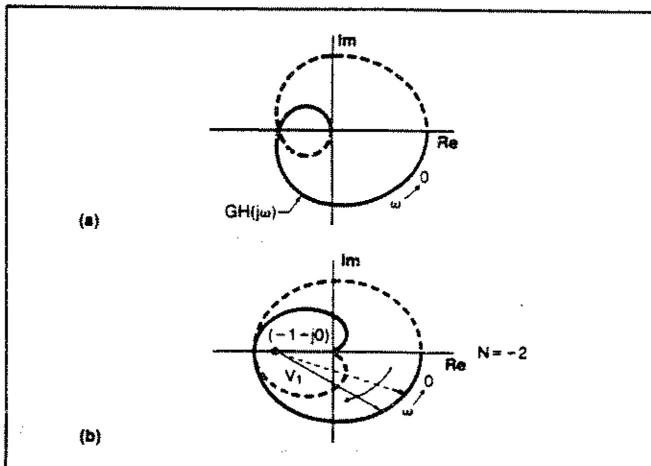


Fig 5—The Nyquist diagram (a), a tool for frequency-domain analysis of a system, *is a plot of the system's open-loop frequency response and its complex conjugate. To determine the number of closed-loop poles with positive real parts, count the number of net rotations of vector* V_1 *in* ***b***.

Closed-loop systems compare input signals with feedback signals in such a way as to maintain a precisely controlled output/input relationship.

vector to the point representing $-1+j0$, allow the head of the vector to trace the entire path created by the open-loop frequency response and its complex conjugate, and then count the net revolutions (N) of that vector, you can determine whether the system has any closed-loop poles with positive real parts **(Fig 5b)**.

The stability criterion actually determines the difference (N) between the number of zeros (Z) and the number of poles (P) of the term $1+GH(s)$, using the equation $N=Z-P$. In systems for which $P=0$ (which is often the case), $N=Z$, which is the number of poles that have positive real parts in the closed-loop transfer function.

The most important aspect of Nyquist's criterion is its usefulness. This tool provides engineers with a way to determine the stability of a control loop by simply measuring the open-loop frequency response. No rigorous mathematical analysis is necessary. Further, Nyquist's criterion allows you to determine the stability of a control loop before closing the loop, thus avoiding possible damage to your system from reactions caused by instability.

The unity-feedback case

For systems with unity feedback ($H(s)=1$), the Nyquist diagram **(Fig 6)** provides a technique for directly calculating the closed-loop frequency response, $G(j\omega)/[1+G(j\omega)]$, from its measured open-loop frequency response $G(j\omega)$. In this case, two vectors are projected—one from point $-1+j0$ (V_1) and the other from the origin (V_2)—to extend to the curve $G(j\omega)$.

V_1 thus represents the term $1+G(j\omega)$, and V_2 represents $G(j\omega)$. The vectors' heads are placed on the open-loop frequency response. You can use simple vector algebra (V_2/V_1) to calculate the input/output relationship of the closed-loop system at the corresponding frequency.

As an alternative to calculating the ratio of the vectors at many frequencies, graphical tools called magnitude contours and phase contours were devel-

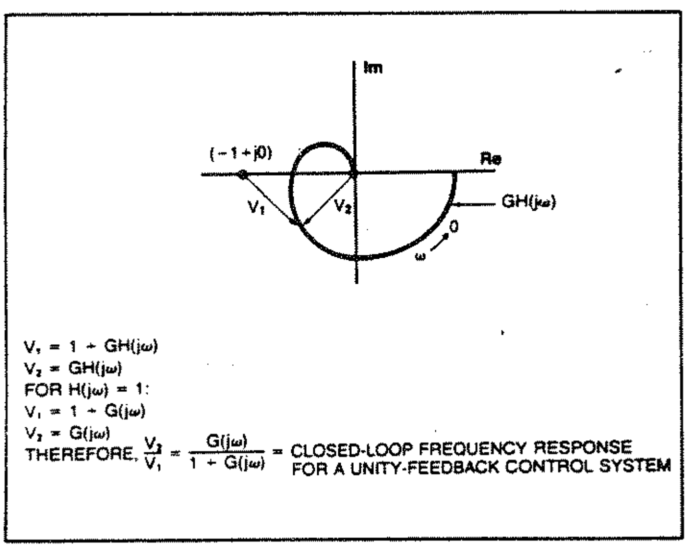


Fig 6—To evaluate unity-feedback closed-loop frequency response *from open-loop data, you use vector algebra with a Nyquist diagram. The ratio V_2/V_1 is the output/input response of the closed-loop system at any frequency.*

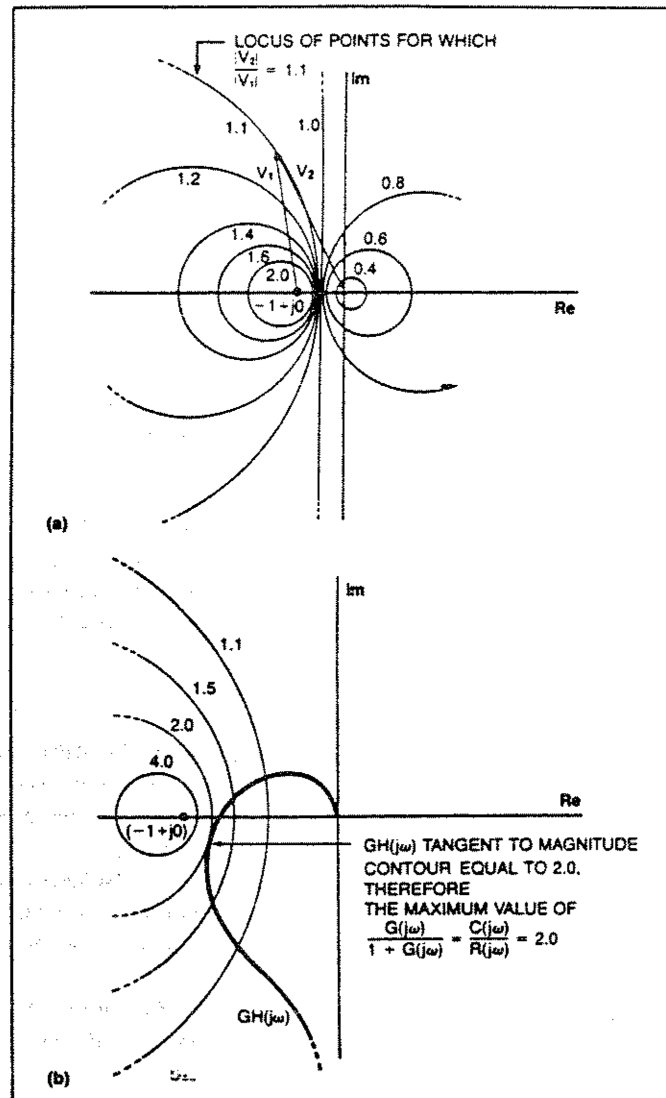


Fig 7—Loci of constant transfer-function magnitude, *or magnitude contours **(a)**, form a family of circles. By plotting an open-loop transfer function against the contours **(b)**, you can identify the gain of the closed-loop function by its intersection with the loci.*

An understanding of classical measurement methods for control systems aids you in using dynamic signal analyzers for design, modeling, analysis, and test.

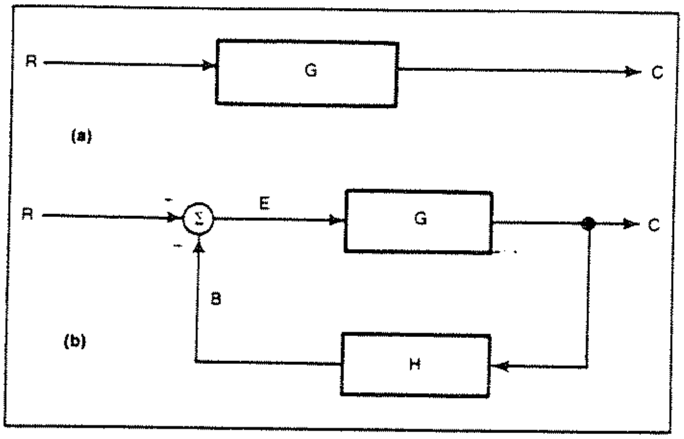


*Fig 1—**The two basic configurations for a system** are the open-loop (a) and closed-loop (b) configurations. In an open-loop system, variations in the output go undetected; in a closed-loop configuration, feedback from the output maintains the desired output characteristics.*

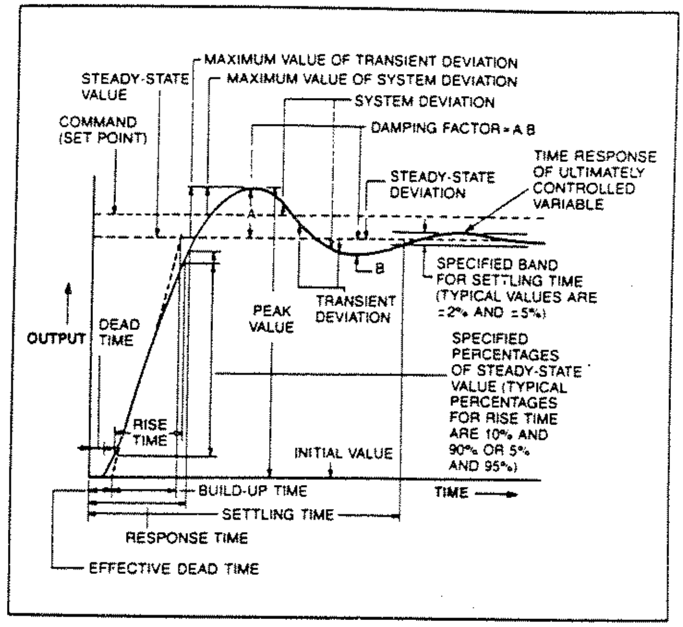


*Fig 2—**This output waveform, a typical response** of a system to a step forcing function, can help you evaluate the stability of the system. The overshoot and ringing are good indicators of the system's stability. Settling time to within a given error band is an important parameter for linear amplifiers and D/A converters.*

junction without any modification, transfer function H will equal one. A system in which H is 1—called a unity-feedback control system—is the easiest type of system for designers to analyze. These block diagrams will help you model a system. To evaluate the system's performance, however, you'll need criteria by which to judge it.

You need to know, for example, how quickly the system achieves the desired output level and whether the system can maintain that level with little or no variation. Control-system designers' need for this basic information led to the first serious study of negative-feedback control systems and ultimately to the development of classical linear control theory.

The first attempts to analyze systems, which were speed regulators on steam engines, took place in the 19th century and involved the use of differential equations. To make the equations as easy as possible to solve, engineers selected a small group of standard inputs that were easy to express mathematically. Chief among these inputs was the step forcing function. Not only was this function simple to express, but it was easy to implement physically, so it permitted the engineer to compare analytical results with measured results.

By using information derived from this step-response technique, you can measure both the speed with which a system reacts to a change in input (rise time) and the degree to which the system temporarily exceeds the desired output level (maximum overshoot). The information also indicates settling time, which is the time it takes for a system to reach a new output level within a given error band (**Fig 2**).

A system's settling-time spec indicates the relative stability of that system. For example, a system that oscillates around the steady-state value for long periods of time is considered to be less stable than a system whose oscillations die out relatively quickly. A system whose response oscillates indefinitely at either a constant or an increasing amplitude after the system has been excited by a step input is considered to be unstable (**Fig 3**).

Although measuring time-domain responses allowed engineers to verify system models and obtain useful, easily extracted performance information, this method provided few clues to how they could improve a system's performance. Further, as systems became more complex, the solution of the integrodifferential equations used to model them grew correspondingly difficult. Solving these equations became somewhat easier in 1899, however, when Oliver Heaviside introduced partial-fraction expansion techniques. Heaviside's discovery made it possible to use Laplace transforms to simplify the solution of large differential equations.

The Laplace transform

The Laplace transform is the integral of the product of the variables f(t) and e^{-st}. The variable s is complex; it has a real component (σ) and an imaginary component (jω). The transformation of a system's transfer function results in a ratio of polynomial expressions in s. In this format, many of the time domain's complex calculations become simple algebra problems.

Of particular interest to engineers are values of s that would cause either the ratio's numerator or denominator to equal zero. Values that set the numerator to zero force the function represented by the ratio to equal zero in the Laplace domain. These values of s are called zeros. Values of s that cause the denominator to equal zero force the function represented by the ratio to equal infinity in the Laplace domain. These values are called poles. Poles are of special significance: When you transform a function back into the time domain to predict the

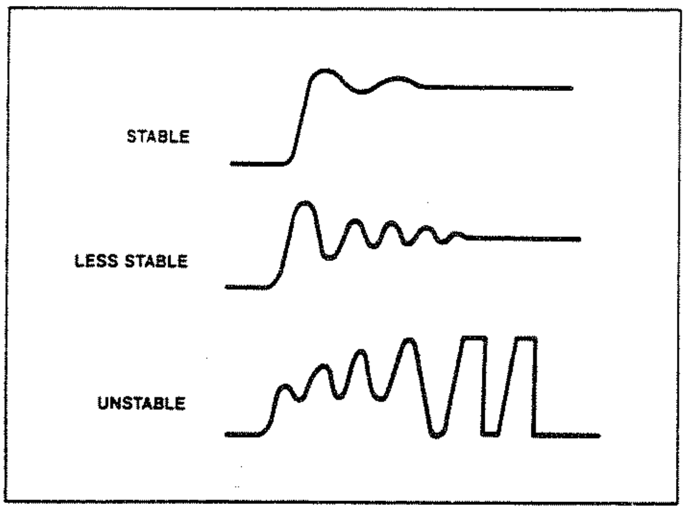


***Fig 3—A system's settling characteristic in response to a step-function input** shows how stable the system is. The more ringing there is around the waveform's final value, the less stable the system is. In an unstable system, the ringing never stops, but instead builds up to oscillation.*

system's response, the real part of the pole determines the exponents of the response. If an imaginary part of the pole exists, it becomes a frequency component of at least one term of the response.

In addition, you can use the real part of the poles to determine the system's stability without having to transform the calculations back into the time domain. Poles with positive real values indicate positive exponents in the time domain; therefore, they indicate instability of the control loop.

These Laplace-transform methods were sufficient to solve control-system problems until the early part of the 20th century. With the development of the vacuum tube and large electronic systems, however, the computational aid of the Laplace transform became inade-

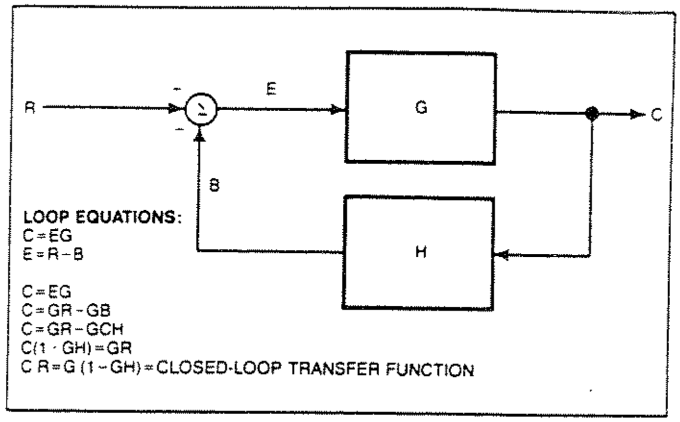


***Fig 4—Simple multiplication, addition, and subtraction** yield the transfer function for a closed-loop control system. The model applies equally well to locomotives and operational amplifiers.*

quate for system evaluation. In particular, Harold S Black's use of negative feedback in electronic-amplifier design in the late 1920s spurred engineers to look for more powerful analytical tools.

From Laplace to Nyquist

In the early 1930s, Harry Nyquist discovered that the solution to the problem of analyzing complex systems lay in the frequency domain. His analysis was built upon the extension of several familiar concepts. By expressing the elements of the control loop as transfer functions (ie, the Laplace transform of the output divided by the Laplace transform of the input), simple algebra could provide an expression for the input/output relationship of a system.

By using the block diagram in **Fig 1** and letting C, E, R, and B represent the Laplace transforms of the signals appearing at the lettered points in the system, you can express the input/output relationship of a closed-loop system as $C/R=G/(1+GH)$ **(Fig 4)**. This expression is known as the closed-loop transfer function. Reduced to its simplest form, the expression is itself a ratio of polynomials in s.

From previous work in the Laplace domain, engineers already knew that the poles of this closed-loop transfer function would determine the stability of the system. If you look for the closed-loop poles (ie, values of s that force $1+GH$ to zero), you see that the GH term contains all the information regarding the poles' whereabouts (see **box,** "Polynomials within polynomials"). Therefore, the key to determining the stability of a system is knowing whether any of the values of s that make GH equal −1 (ie, the closed-loop poles) have positive real parts.

Graphical-analysis techniques allow you to use a negative-feedback system's open-loop response to predict the system's closed-loop response.

measure on a Bode plot. Open-loop bandwidth indicates how fast a system can react to a change in input; this measurement is analogous to rise time in the time domain. Specifically, open-loop bandwidth is the span between 0 Hz and the frequency at which the open-loop response has unity gain (0 dB). The greater the open-loop bandwidth is, the faster the system will react.

No graphical technique using a Bode plot exists for directly calculating the closed-loop frequency response of unity-feedback systems. Although such a technique (the use of magnitude and phase contours) does exist for the Nyquist diagram, this diagram's linear scales would seldom accommodate both the range of gain values produced by a system and the level of detail about the unity-gain point required to determine stability.

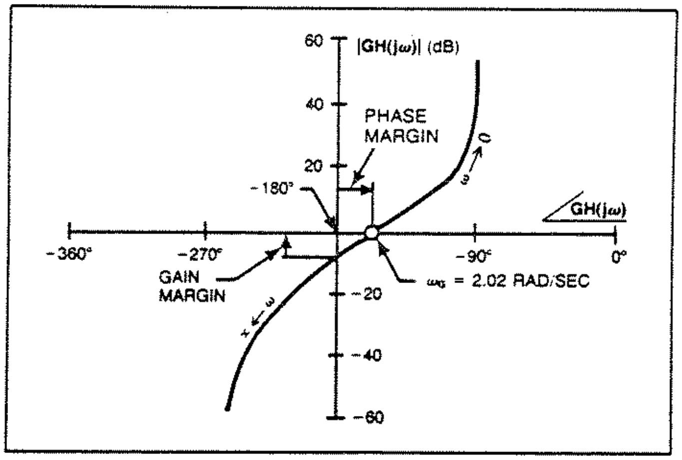


***Fig 10—By plotting logarithmic gain against linear phase** in a Nichols diagram, you can calculate graphically a unity-gain system's closed-loop frequency response from the open-loop response.*

The Nichols diagram, a third graphical analysis tool, was developed specifically to accommodate the graphical calculation of the closed-loop frequency response of a unity-feedback system from the system's measured open-loop frequency response. The Nichols diagram uses both the single-plot concept of the Nyquist diagram, in which frequency is a varying parameter (but not a coordinate), and the logarithmic gain and linear phase scales of the Bode plot. Thus, the Nichols diagram is a plot of logarithmic gain versus linear phase as a function of frequency, as **Fig 10** shows.

Because of the wide range of gains represented by the logarithmic scale, however, the open-loop frequency response of almost any system could be plotted on a standardized grid. It was practical, therefore, to superimpose a second grid (of magnitude and phase contours)

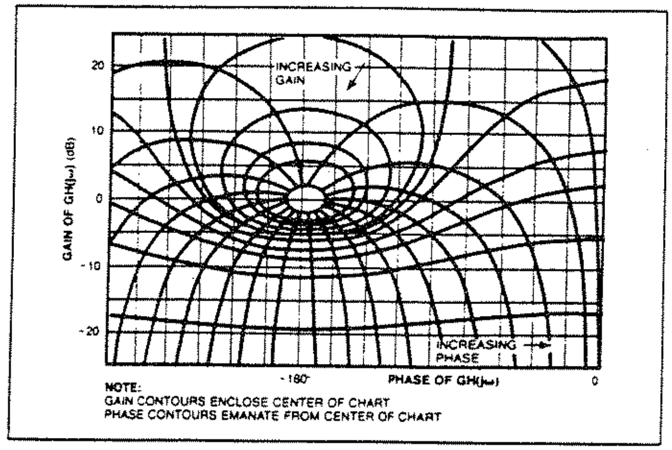


***Fig 11—Magnitude and phase contours are superimposed** on a Nichols diagram, forming a Nichols chart. The chart allows you to make a quick estimate of the closed-loop response of a unity-feedback system.*

on the Nichols diagram for calculating the closed-loop frequency response. The resulting Nichols chart (**Fig 11**) rapidly became a standard graphical tool for quickly estimating the closed-loop frequency response of a unity-feedback system. Unfortunately, like the Nyquist diagram, the Nichols chart doesn't make it easy for you to combine frequency responses, nor does it provide graphical tools for linking system models and frequency responses. **EDN**

5959-5738
Printed in U.S.A.

EDN

Designer's Guide to:
Linear control-system theory—Part 2

Dynamic signal analyzers simplify measurement of linear control systems

Reprinted from EDN April, 1986

Dynamic signal analyzers simplify measurement of linear control systems

Advanced dynamic signal analyzers (DSAs) give designers a wide choice of techniques for measuring a system's open-loop frequency response. This article, part 2 of a 3-part series, considers the effect of DSAs on the graphical measurement techniques of linear control-system theory. Part 1 of the series presented an overview of classical linear control theory. Part 3 will explore the expanded role of DSAs in the control-system design process.

Steve Asbjornsen and Owen Brown,
Hewlett-Packard Co

You can use a variety of graphical techniques to analyze a negative-feedback, closed-loop control system (**Ref 1**). Bode plots, Nichols diagrams, and Nyquist diagrams are three distinct tools that allow you to determine a system's amplitude and phase characteristics (and, therefore, its stability) as functions of frequency. Typical test instruments make use of these tools, and a dynamic signal analyzer (DSA) is no exception—it allows you to view and plot frequency-response data in all three formats.

The internal analysis functions of modern DSAs, however, alter the relative usefulness of these three graphical techniques. Understanding these DSA functions gives you an idea of how a DSA's computational power can expand your test options.

Waveform math and Nyquist diagrams

A DSA's waveform-math capability, for example, limits the Nyquist diagram's usefulness to providing a complete check of a system's stability and a 1-trace representation of its frequency response. The waveform-math utility is a built-in calculator that allows you to add, subtract, multiply, divide, or use any of the other operators shown in **Table 1** to manipulate frequency responses, recorded waveforms, and complex constants.

The Nyquist diagram lets you easily determine the stability of all types of systems, including absolutely and conditionally stable systems. You can also directly calculate the closed-loop frequency response of a unity-feedback control system. However, the diagram doesn't facilitate calculation of composite frequency responses, and its linear scales can't accommodate both adequate gain ranges and acceptable resolution around the unity-gain point. Reading phase margin is more difficult with the Nyquist diagram than with other diagrams, and reading the open-loop bandwidth is

A DSA's built-in calculator uses waveform math to perform arithmetic operations and to manipulate frequency responses, waveforms, and complex constants.

TABLE 1—WAVEFORM-MATH FUNCTIONS IN A DSA

ADD	SQUARE ROOT	MULTIPLY BY jω	T/(1−T)
SUBTRACT	RECIPROCAL	FFT	REAL PART
MULTIPLY	NEGATE	INVERSE FFT	COMPLEX CONJUGATE
DIVIDE	DIFFERENTIATE		LOG DATA

impossible unless the frequency at which the gain becomes unity is recorded on the plot. Finally, you can't use the Nyquist diagram to estimate the transfer function of a system from its measured frequency response, or vice versa.

A DSA's waveform-math utility, however, lets you calculate a system's closed-loop response precisely, in any format, using the equation

$$C(j\omega)/R(j\omega)=G(j\omega)/\{1+G(j\omega)\}.$$

Using waveform math, you can calculate closed-loop frequency response independently of the display format. Linear scales are not a problem when you're using a DSA, because the DSA's display can easily rescale the data. The DSA's marker readouts make the measurement of gain margin, phase margin, and open-loop bandwidth much easier.

Bode plots

DSAs also alter the usefulness of the Bode plot, which designers have traditionally favored because they could use it to estimate composite frequency responses quickly. The Bode plot's logarithmic units offer a large dynamic range of gains, and the plot makes it easy to measure gain margin, phase margin, and open-loop bandwidth. Finally, unlike the Nyquist diagram, the Bode diagram lets you estimate a transfer function from a frequency response and vice-versa.

The Bode plot's major drawbacks are that you have to plot traces for both gain and phase, and that you can't estimate the closed-loop frequency response from the open-loop frequency response. A DSA's waveform-math utility makes it easy to calculate the closed-loop frequency response, however. Further, the DSA's frequency-response-synthesis and curve-fitting functions automate the transition between frequency responses and transfer functions.

The Bode plot is still useful in that it helps you intuitively understand the frequency-response/transfer-function transition. The Bode plot also helps you estimate composite waveforms, and its logarithmic gain units provide both range and resolution.

Although the Bode plot and Nyquist diagram are still useful to designers who perform system analysis on DSAs, Nichols diagrams are not. The Nichols diagram's only advantage over the other diagrams is that it lets you calculate closed-loop frequency responses. Because the waveform-math capability of DSAs solves this problem, it renders the Nichols diagram obsolete.

The root-locus plot

The newest method of making control-system measurements is root-locus analysis, which was developed in the late 1940s and early 1950s by Walter R Evans. All previous methods of analysis had used open-loop frequency response solely to determine whether closed-loop poles with positive real parts existed. These methods yielded no additional information concerning the actual value of s for the poles. The root-locus technique, however, lets you examine the actual values of s for the closed-loop poles graphically, based on the known values of s for the open-loop poles and zeros.

The root-locus diagram could not have been conceived without the development (in the late 1940s) of the s-plane. The s-plane is a 2-dimensional plane that represents all possible values of the Laplace variable s. The plane's ordinate is the imaginary part (ω of $s=\sigma+j\omega$), and its abscissa is the real part (σ of $s=\sigma+j\omega$), of s (**Fig 1**).

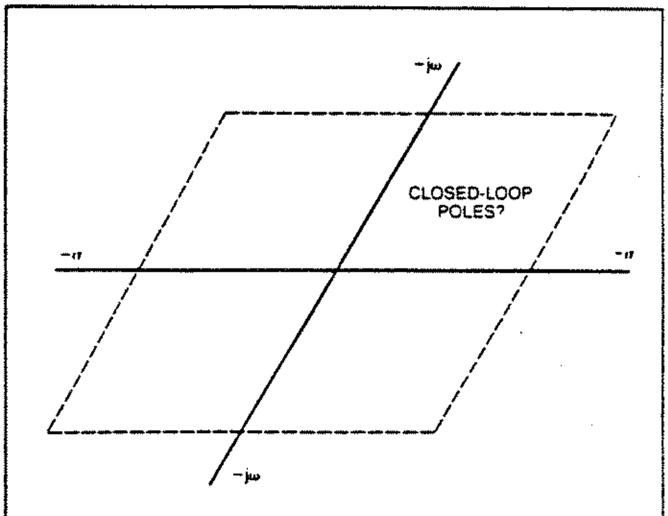


Fig 1—The s-plane represents all possible values of s in two dimensions. This plot allows you to locate the poles and zeros of a closed-loop transfer function and thereby determine the stability of the system characterized by the transfer function.

Each value of s, therefore, has a unique position in the s-plane. If you can determine the poles and zeros of a ratio of polynomials (such as the open- and closed-loop transfer functions of a control system) in s, you can plot the location of these poles and zeros in the s-plane. Any closed-loop pole that exists in the right half of the s-plane represents poles with positive real parts and therefore indicates an unstable system.

In measuring the open-loop frequency response of a system, you're collecting the same data you would if you were evaluating GH(s) for values of s that lie on the positive ordinate of the s-plane ($s=0+j\omega$ for $\omega=0$ to +infinity). From the open-loop information, Nyquist—without using the idea of the s-plane—made the conceptual leap that allowed him to determine whether there were any values of s with positive real parts that solved the equation $GH(s)=-1$. His observation was not an obvious one, to say the least.

Consider a control loop that has been opened so that the open-loop frequency response ($GH(j\omega)$) is measurable. If you alter just the gain of the loop, you won't affect the value of the open-loop poles and zeros. As a result, the open-loop transfer function can be expressed as $GH(s)=KGH(s)$, where K represents a proportional gain constant that's independent of s.

Although varying K has no effect on the position of the open-loop poles and zeros, it can have a tremendous effect on the closed-loop poles. This effect becomes apparent if you substitute KGH(s) in the denominator of the closed-loop transfer function $G(s)/\{1+GH(s)\}$ and solve the equation for the poles. The resulting expression is $1+KGH(s)=0$.

The closed-loop poles, therefore, are values of s that are solutions to the equation $GH(s)=-1/K$. To locate the closed-loop poles, you must, whenever K changes, find new values of s that satisfy the equation $GH(s)=-1/K$. It was this relationship—between the stationary poles and zeros of the open-loop transfer function, and the closed-loop poles that vary with pure gain—that provided the basis for Evans's root-locus technique.

Using the root locus

The root-locus technique plots the open-loop poles and zeros in the s-plane. You can obtain the open-loop pole and zero locations from a mathematical derivation of the open-loop transfer function or by using Bode's techniques to extract the transfer function from a measured frequency response. If you plot the open-loop poles and zeros on the s-plane (**Figs 2a and 2b**), you can

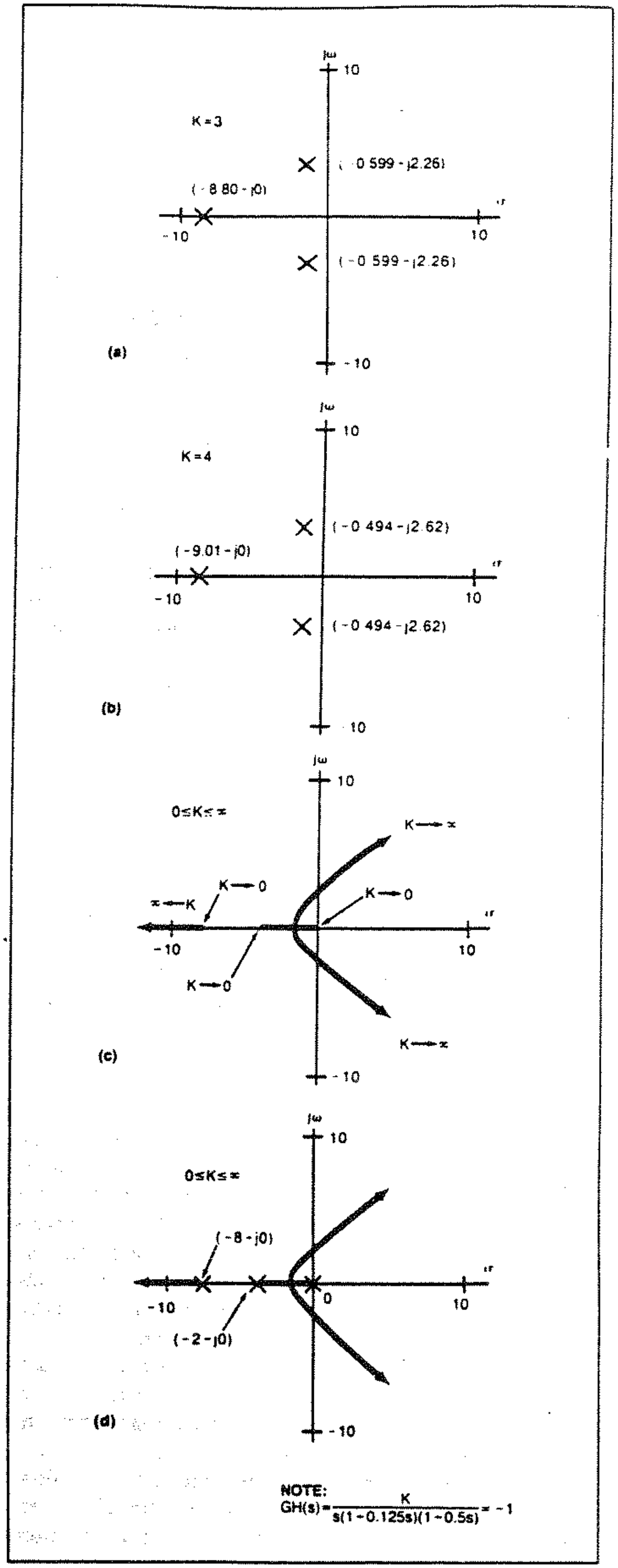


***Fig 2—A powerful graphical technique, the root-locus plot,** allows you to determine the location of poles in a closed-loop system without actually measuring the closed-loop response. This method depicts the migration of poles as the system's frequency-independent gain (K) varies.*

use Evans's graphical techniques to draw a trace that represents the migration of the closed-loop poles (or root loci) as the frequency-independent gain varies (**Figs 2c** and **2d**).

The advantage of this technique lies in its ability to give the actual location of closed-loop poles without actually measuring the closed-loop response. However, using graphical techniques to determine the root loci doesn't give you the actual value of the frequency-independent gain (K) at any particular point in a locus. You must return to the equations and calculate the closed-loop pole locations for several values of K until you discover the value of K that corresponds to some point on a locus.

The root-locus technique also requires that you know the number and location of the open-loop poles and zeros before you can estimate the position of the closed-loop poles. The technique is, therefore, less flexible than the Nyquist or Bode diagrams, which let you predict stability, measure performance, and obtain design information, but allow you to measure only the open-loop frequency response. The root-locus method, however, provides you with more information during the initial design process, and it's better suited to the design of the complex compensation networks typically associated with complex systems. Also, because you know the position of the closed-loop poles, you can derive the time-domain response for a given value of gain (K).

Almost all linear control-system analysis, and much of the subsequent designs, depends on obtaining an accurate estimate of a system's open-loop characteristics, either in the form of a frequency response $GH(j\omega)$ or the closed-loop transfer function $GH(s)$. No matter how these open-loop characteristics are expressed, designers must always perform the actual physical frequency-response measurements, whether to help construct system models or to verify them.

Measurement techniques

In the past, engineers had to choose between two types of analyzers for making low-frequency control-system measurements. They could choose either the classic frequency-response analyzers (FRA), which provide swept-sine measurements, or fast-Fourier-transform analyzers (FFTA), which can measure a whole spectrum in one measurement.

The two analyzers have different advantages and disadvantages. For example, although the FFTA has the potential for faster measurement times, it entails complex set-up procedures. And although the FRA is familiar to engineers, who understand its swept-sine method of measurement, its measurement times are slow. However, because DSAs offer both measurement techniques, designers no longer have to accept the tradeoffs that accompany choosing an FFTA or an FRA.

Frequency-response analyzers

FRAs operate in much the same way as do heterodyne network analyzers, and they're limited to taking measurements at low frequencies. They generally possess two channels, each of which uses a discrete Fourier transform to emulate a single bandpass filter. The Fourier integration time controls the filter's bandwidth to values in the low microhertz range, and an integrated sine-wave source (**Fig 3**) synchronizes the filter's center frequency.

A stimulus signal from the FRA drives the device under test. The analyzer's two channels connect to the input and output of the device, and the signal each channel receives undergoes comparison with the stimulus signal as a function of the discrete Fourier transform. The result is a complex value containing the magnitude and phase (with reference to the stimulus signal) of the measured signal.

The FRA then compares the two channels' results, deriving the gain and phase-shift relationships between the two channels' signals. This process occurs several

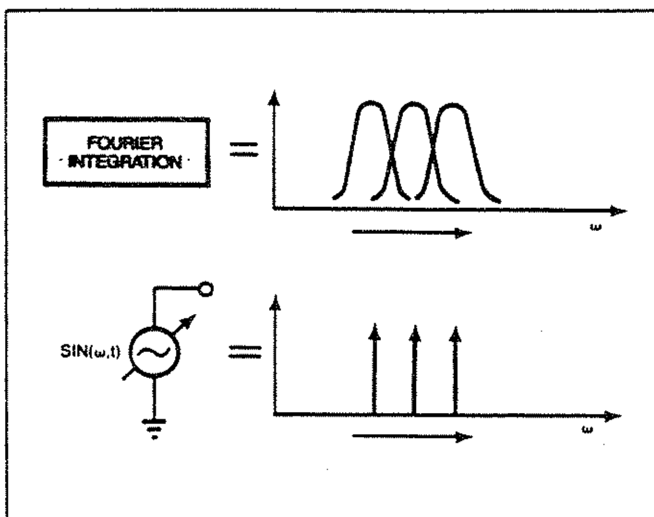


***Fig 3—The integration time of a Fourier transform** controls filter bandwidth, and a sine-wave signal source sets the filter's center frequency. Each discrete Fourier transform emulates a single bandpass filter.*

A root-locus diagram allows you to examine the values of s for a system's closed-loop poles, based on the known values of s for the open-loop poles and zeros.

times between the start and stop frequencies being analyzed, thereby producing a series of discrete gain and phase values. When you connect these points graphically, you obtain the gain and phase curves of the frequency response. Note that this measurement process has two implementations, and the difference between them can be important to the designer who uses computer-aided analysis.

The primary distinction between the two implementations lies in the sources. An FRA whose source sweeps continuously can integrate the signal while the source is actually sweeping. Each integration period, therefore, covers a small part of the total measurement span. The result of that integration is then available at the end of each integration period.

This continuous-sweep technique creates a potential ambiguity between the phase and magnitude values for the displayed frequency, and the exact frequency at which they occurred. The ambiguity becomes especially serious when the integration period covers large frequency spans. Because the integration period is typically fixed, you can generally minimize this problem by reducing the sweep speed—and therefore the frequency span covered—during integration. The ambiguity won't interfere with graphical analysis, but it can create difficulties in computer analysis.

Sweep-and-dwell sources

The alternative to the continuous-sweep implementation is a sweep-and-dwell sine-wave source. In this type of analyzer, the sine-wave source dwells at a discrete frequency during the integration process and then performs a phase-continuous sweep to the next analysis frequency.

Because a sweep-and-dwell analysis occurs at a discrete frequency, the phase and gain analyses apply only for the frequency point at which the measurement was made; therefore, no ambiguity exists. DSAs incorporate this sweep-and-dwell form of swept-sine analysis, which optimizes the accuracy of their integrated computer-aided-analysis functions.

One possible drawback to the sweep-and-dwell technique is that the analyzer might miss valuable information between measurement points. However, by simply decreasing the sweep rate of these analyzers, you increase the number of measurement points between the start and stop frequencies and provide better resolution.

Newer analyzers offer an autoresolution function that monitors the gain and phase shift between measurement points and automatically adjusts the resolution (ie, sweep rate) during the sweep, thereby preventing the loss of valuable data. This function can also minimize total sweep time by increasing the sweep rate in portions of the frequency response that are relatively flat in both gain and phase.

FFT analyzers

Fast-Fourier-transform analyzers (FFTAs) are similar to FRAs in that they use a type of Fourier transform to achieve narrow analysis bandwidths. Their method of signal generation and use of two channels to compare a device's input and output are also the same as those of the FRAs. However, instead of emulating a single bandpass filter and tracking it over the spectrum of interest, FFTAs emulate hundreds of bandpass filters (Fig 4) and provide complete coverage of an entire spectrum in one integration period. FFTAs can usually perform measurements much more quickly than can FRAs.

In addition to the increased number of analysis bands, the FFT process can also use a wide range of stimulus signals. They typically use stimulus signals (such as random noise) that provide energy over the entire analysis span, thus taking full advantage of the analysis power.

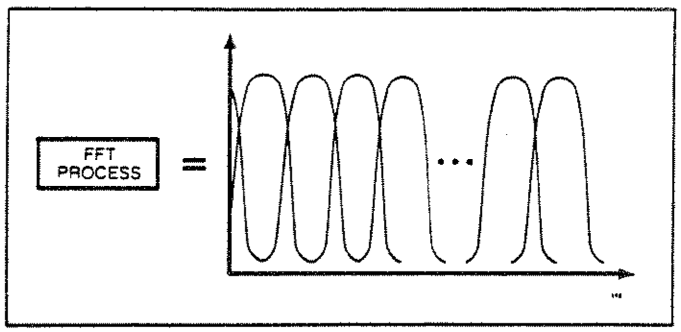


Fig 4—FFT analyzers emulate hundreds of bandpass filters. *One integration period, therefore, provides complete coverage of an entire spectrum. The FFTA can use a wide variety of stimulus signals, including random noise.*

FFTAs and FRAs use different methods to reduce measurement noise. If you don't know what the differences in the methods are and how they affect the measurement process, you can very easily misuse an FFT analyzer. Designers lacking this information have sometimes concluded—mistakenly—that FFT analyzers can't make control-system measurements.

Although the term "swept-sine analysis" describes the FRA's stimulus signal, it doesn't describe the

The sweep-and-dwell sine-wave stimulus is better than any other type of stimulus for measuring noisy systems.

FRA's unique analysis process comprehensively (heterodyne analyzers also use a swept-sine stimulus). The term "swept Fourier analysis" (SFA) describes the FRA's measurement process more specifically. The differences between FFT and SFA measurement processes lie mainly in their stimulus signal, single- versus multiple-band analysis, and noise-reduction techniques.

Stimulus signals in SFAs and FFTs

The SFA measurement process uses a swept-sine-wave stimulus, and the FFT process uses stimuli that produce energy at all the analysis frequencies within a single integration period. When you're measuring extremely noisy systems, the type of stimulus itself can have a profound effect upon the measurement.

The sweep-and-dwell sine-wave stimulus is better than any other type of stimulus for measuring noisy systems, because the power of the stimulus is concentrated at one discrete frequency. This concentrated-power approach automatically provides the best possible signal-to-noise ratio without any signal processing. A random-noise stimulus, on the other hand, must distribute its energy over a wider bandwidth, providing less power at any one discrete frequency than would a dwelling sine wave (ie, its power spectral density is much lower than a sinusoid's) (**Fig 5**).

A random-noise stimulus also has advantages, however. One of the key strengths of this type of stimulus is that it provides a linear estimate of the operation of a nonlinear system. For example, many systems experience changes in their frequency response relative to the drive level or relative to the direction of a sine-wave sweep. Random noise, which has no sweep direction and has random amplitudes at all frequency components, provides an average of the drive-level and sweep-direction effects, so it usually provides a good approximation of a system's operation.

In situations in which initial measurements indicate that the energy level of the random-noise stimulus is too low, you can improve the relative power spectral density of the stimulus by reducing the frequency span of the measurement (if the analyzer uses a band-limited random-noise source). However, to cover the original frequency span of interest, you must take more measurements.

Single- vs multiple-band analysis

The SFA's single-filter measurement process is slower than the FFT process, which provides hundreds of filters. However, the use of a single filter does have its advantages. If you use a single filter, you can make all the signals produced by distortion products (such as harmonic distortion and intermodulation distortion) lie outside the analysis bandwidth of a single filter, thus removing the products from the measurement.

By increasing the Fourier integration time, you can always reduce the filter's bandwidth to exclude distortion products (**Fig 6**). The only time you can't remove a disturbance signal is when a spur at a fixed frequency occurs at exactly the same frequency as that of the SFA's stimulus. Because of its many filters, the FFT process can be affected by distortion products, depending on the stimulus used.

For example, if you use a sine-wave stimulus in a

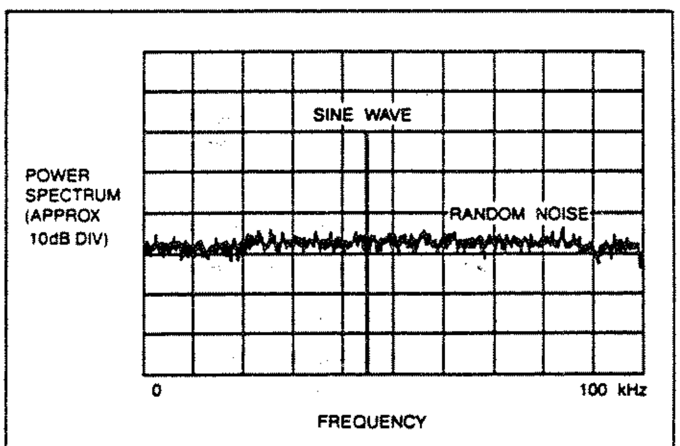


Fig 5—Power-spectrum density for a random-noise signal *is much lower than that for a sine wave, as this plot shows. The random-noise signal distributes its energy over a much wider bandwidth than does a sine wave.*

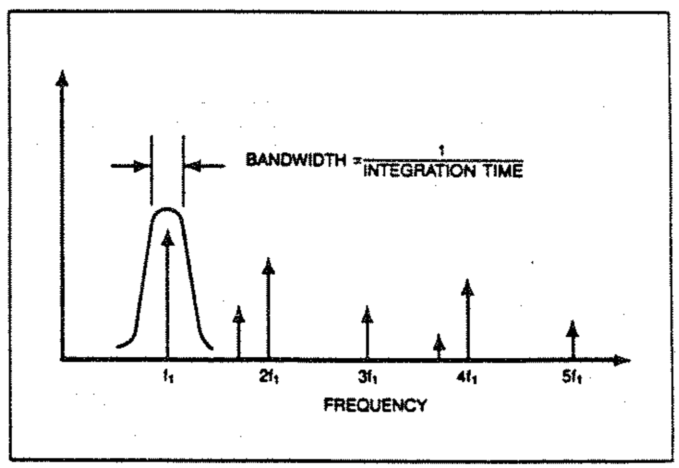


Fig 6—Reducing the analysis band of an SFA's measurement process *excludes distortion products from the measurement. The analysis band is inversely proportional to the Fourier integration time.*

A DSA combines the advantages of classic frequency-response (swept-sine) analyzers and fast-Fourier-transform analyzers.

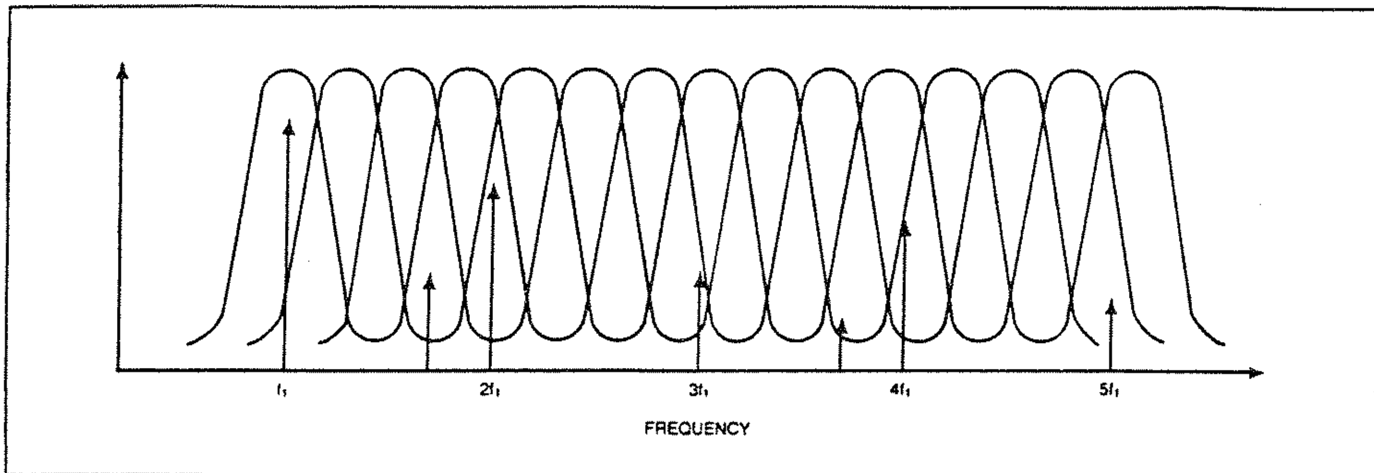


Fig 7—A sine-wave stimulus in a measurement using the FFT process can cause distortion products to appear within the filters produced by the transforms. The products would thus be recorded as part of the system's response.

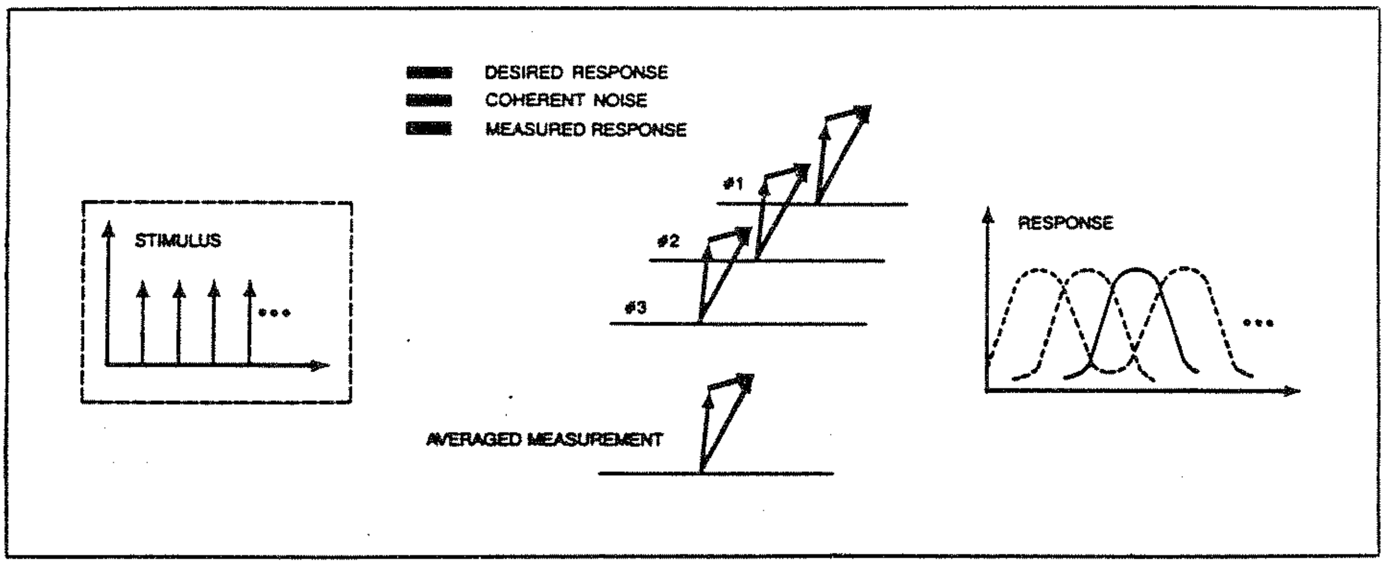


Fig 8—Periodic stimuli allow harmonics to maintain a constant phase relationship with desired signals. It's impossible, therefore, to average these components to zero, thereby reducing the effect of the nonlinearity on a measurement. A random stimulus eliminates this distortion-induced problem; averaging causes the nonlinear portion of the response in each filter to dwindle to zero.

system that produces harmonic distortion, the distortion products can appear within one or more of the FFT's filters and be recorded as part of the system's response (Fig 7).

Distortion products can also affect the FFT process when you use a pseudorandom signal as a stimulus. You can characterize this pseudorandom signal as a summation of discrete sine waves, each of which is tuned to the center frequency of a unique filter. If you separate the response in each filter into the portion of the response that results from the intended stimulus (the desired response) and the portion of the response that arises from a harmonic product of a lower frequency, you'll never see a change in the relationship between the desired response and the distortion product from measurement to measurement. Therefore, even if you were to average the results of several measurements, you wouldn't reduce the effect of the nonlinearity on the measurement (Fig 8).

A random stimulus, however, eliminates this distortion-induced problem by letting the nonlinear portion of the response in each filter decay to zero with averaging, even if the nonlinearity is a fixed spur at the center frequency of a filter. A swept-sine-wave stimulus

An FFT analyzer uses two channels to compare input and output and emulates hundreds of bandpass filters to provide complete coverage of an entire spectrum.

doesn't permit such averaging. The explanation of distortion phenomena leads directly to the topic of noise reduction.

Noise reduction in SFAs and FFTs

The noise-reduction process of SFAs is fairly straightforward. If you increase the integration time in each channel, the analysis bandwidth in each channel becomes smaller and smaller while remaining centered on the frequency of the stimulus. As the bandwidth becomes smaller, the noise power (of the measured system) within the filter lessens. Moreover, any spurs close to the center frequency become located farther down the stop band of the analysis bandwidth until they receive sufficient rejection.

The only type of distortion that can't be rejected is a spur that has the same frequency as the stimulus and that maintains a constant phase relationship with the stimulus. In this case, the spur is said to be coherent with the stimulus. A key aspect of this type of noise reduction is that the noise in each channel is reduced before the gain and phase relationship (ie, the frequency response) between channels is calculated.

The FFT process makes two types of noise reduction available: time averaging (a form of linear averaging) and power-spectrum averaging. Time averaging is very similar to the SFA's averaging process in that it improves the S/N ratio of the signal in each channel before the frequency response is calculated. The time-averaging method gathers the samples of the signal normally considered by the FFT into blocks of data called time records, and then averages the time records.

To keep averaging from reducing the signal of interest, you must make sure that the signal is a periodic one (such as the pseudorandom signal mentioned above) and that the phase of the signal is the same in each time record. You must also supply a trigger signal to the analyzer to indicate when data collection should begin.

Power-spectrum averaging

The second, and more commonly used, form of noise reduction in FFT measurements is power-spectrum averaging. The fundamental difference between this technique and time averaging is that relatively little

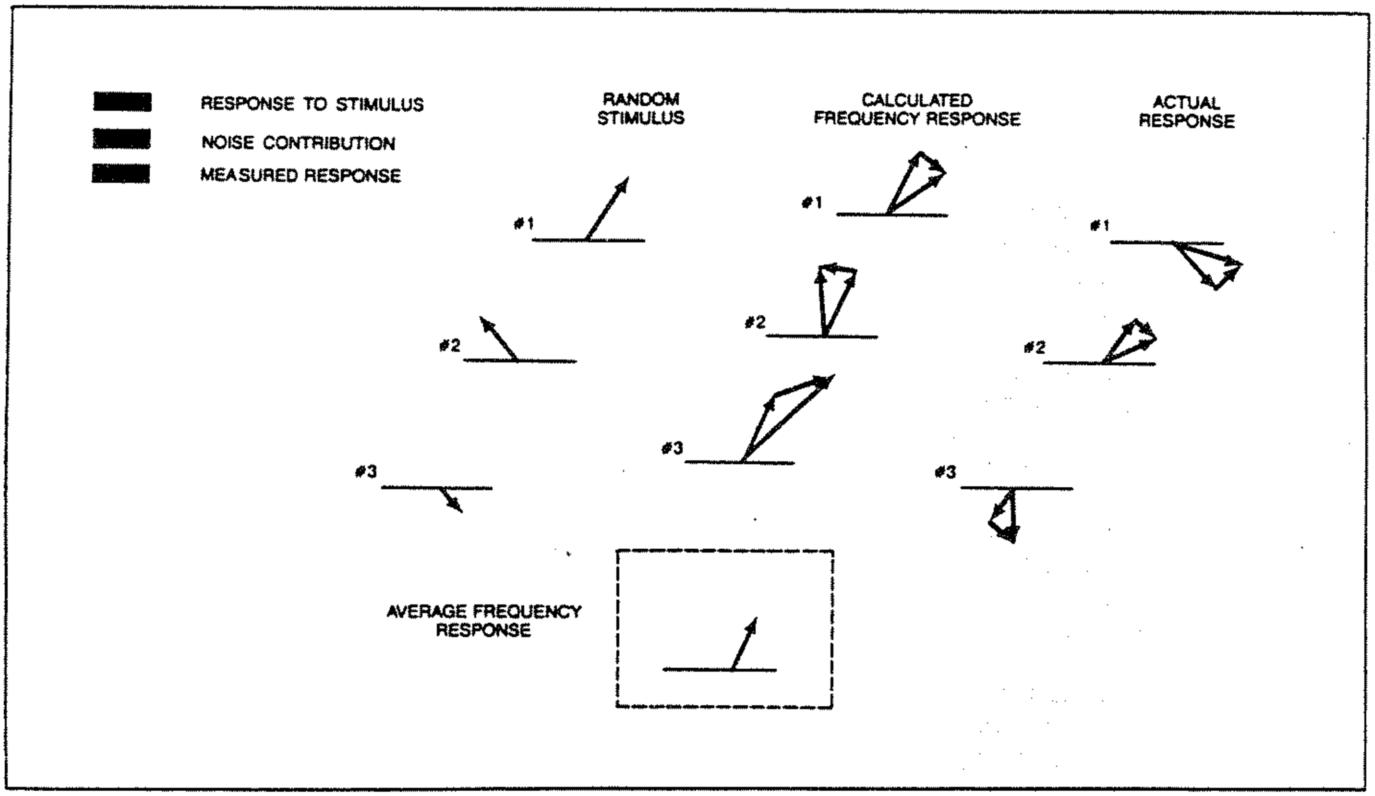


Fig 9—If you average frequency-response calculations, *you can use random noise as a stimulus. The stimulus-response relationship remains constant, while the noise contribution in both channels averages to zero. The averaged frequency response thus converges in a linear representation of the system's frequency response.*

If you don't understand the differences in the ways an FFTA and an FRA effect measurement-noise reduction, you could misuse the FFTA.

noise reduction occurs in each channel. Instead, the method removes noise by averaging frequency-response data from each measurement.

The benefit of this noise-reduction technique is that it doesn't matter how much the stimulus signal differs from one measurement to the next, as long as the gain and phase relationship from measurement to measurement remains the same. You can thus use random noise in an averaged measurement.

For example, assume you allow the random-noise stimulus to maintain the same frequency components as the pseudorandom-noise signal discussed earlier, but let the phase of the discrete sine waves vary randomly (a poor representation of random noise, but useful for this example). In this case, the stimulus signal within each filter will have a different phase orientation in each measurement.

If you were to examine a filter whose output is composed of a linear response to the intended stimulus signal and a nonlinearity-induced harmonic product, you'd see a change from measurement to measurement in the relationship between the linear response and the harmonic. The disparity exists because the phase of the harmonic's fundamental and the intended stimulus would have changed between measurements.

If, over several measurements, you examine a vector representing the computed frequency-response data for each measurement, you'll see that the distortion product appears as a vector that rotates about the end of a stationary frequency-response vector (**Fig 9**). When you average several frequency-response vectors, the contribution from the distortion product falls to zero. The averaged frequency-response vector thus gives you the best linear estimate of the device's frequency response. If you use a random-noise stimulus in this type of averaging scheme, you'll find that even a spur at the center frequency of a filter would be noncoherent with the stimulus and would average to zero.

Although power-spectrum averaging, combined with a random-noise stimulus, reduces the effects of all forms of distortion products from a measurement, you wouldn't benefit from using power-spectrum averaging with a periodic stimulus. Using a periodic stimulus would allow distortion products to be coherent with the stimulus, so they'd be unaffected by averaging. Further, certain control-system measurements don't allow the use of power-spectrum measurement.

The FFT analyzer is always better than a swept-frequency analyzer for measuring a basically linear system with poor to good S/N conditions. Both analyzers (FFTA and SFA) will provide the same response, but the FFT process will provide it much more quickly. The SFA, on the other hand, gives you the best possible S/N ratio, so it's more suitable for use in difficult measurement situations. Having both techniques available is clearly preferable, as in a DSA, so you can use them to handle different measurement problems. **EDN**

5959-5739
Printed in U.S.A.

EDN

Designer's Guide to:
Linear control-system theory—Part 3

Analyzers aid in the design of closed-loop systems

Reprinted from EDN May, 1986

Analyzers aid in the design of closed-loop systems

You can use the various functions of a dynamic signal analyzer to derive the open-loop frequency response of a system from measurements taken at various points in the system. This article, Part 3 of a 3-part series, discusses how to use DSAs to develop and model a control system. Part 1 of the series presented an overview of classical linear control theory, and part 2 considered the role of DSAs in control-system measurement.

Steve Asbjornsen and Owen Brown,
Hewlett-Packard Co

By manipulating the classical graphical-analysis tools of linear control theory, a dynamic signal analyzer (DSA) can greatly assist you in designing and modeling stable closed-loop, negative-feedback control systems. You can use a DSA's internal waveform-math functions to calculate the frequency response of a closed-loop control system by measuring its open-loop frequency response (**Refs 1 and 2**). By analyzing a closed-loop system's open-loop response, you can tell whether the system is stable or unstable, and you can design compensation networks to stabilize an inherently unstable system.

DSAs allow you to use three separate techniques for measuring the open-loop frequency response of a closed-loop system: the loop-open direct method, the loop-closed direct method, and the loop-closed calculated method. In the loop-open direct method (**Fig 1**), which is defined by the expression $B(j\omega)/E(j\omega)$, you open the loop by removing the summing junction from the system. You then inject a stimulus signal at point $E(j\omega)$ and measure the response at point $B(j\omega)$.

The ratio $B(j\omega)/E(j\omega)$ provides the gain and phase characteristics of $GH(j\omega)$, the open-loop frequency response. This technique lets you determine the stability of the loop before you close it. Before taking this measurement, however, you must take steps to avoid four problems: overdriving the system, changing loading conditions, system saturation, and loss of operation.

Overdriving the system

When you stimulate a high-gain system, you must be careful not to overdrive the system, that is, to exceed any part of the system's maximum operating range (a situation that might arise, for example, when a sinewave stimulus sweeps through a resonance). If you exceed this range, you could introduce nonlinearities into the measurement. Furthermore, if the stimulus contains high-energy components, you could damage the system.

Fortunately, you can often avoid overdrive by using a DSA's monitoring functions. The monitoring functions provide source-level control that can vary the stimulus level to maintain a specified input level to either channel of the analyzer.

You must pay attention to the system's loading conditions. To obtain a measured frequency response that's an accurate representation of GH(jω), you must make sure the open-loop system is loaded with the same impedances during testing that will exist when the loop is closed.

You must also avoid system saturation, the condition in which the output at B(jω) gets stuck at the maximum output level when you open the control loop. Such a condition is often caused by the reaction of extremely high loop gains to very small dc offsets—or to integrating components that naturally accumulate stray dc levels—in the absence of the countering effects of negative feedback. When system saturation occurs, you must abandon the loop-open direct technique.

Finally, you must avoid loss of operation (ie, system disruption to the point of failure), which can occur when you open the system's control loop. For example, consider a disk drive's read/write-head-positioning system and the magnetic interface between the head and a prerecorded track on the disk platter. If you open the loop, you won't be able to maintain the interface so that it stays within normal operating conditions. If you stop the disk so that you can position the head over a track on the disk, the magnetic interface will disappear. If you rotate the disk, the slightest off-center condition will cause the head to skip across several tracks. Even the weakest stimulus signal will cause a similar effect. You simply can't test such a system with the loop open.

For a system that's suitable for loop-open direct testing, you should preferably use the DSA's FFT function with a nonperiodic stimulus signal. The FFT function works well because it yields good signal-to-noise performance and because, when you use a nonperiodic stimulus, the DSA allows you to average out distortion products. Further, the FFT function's speed can greatly reduce the total measurement time. When you use a nonperiodic stimulus to reduce the effects of system noise, you must use power-spectrum averaging.

The loop-closed direct method, which is defined by the transfer function Y(jω)/Z(jω), uses the connection shown in **Fig 2.** The technique is a common one for

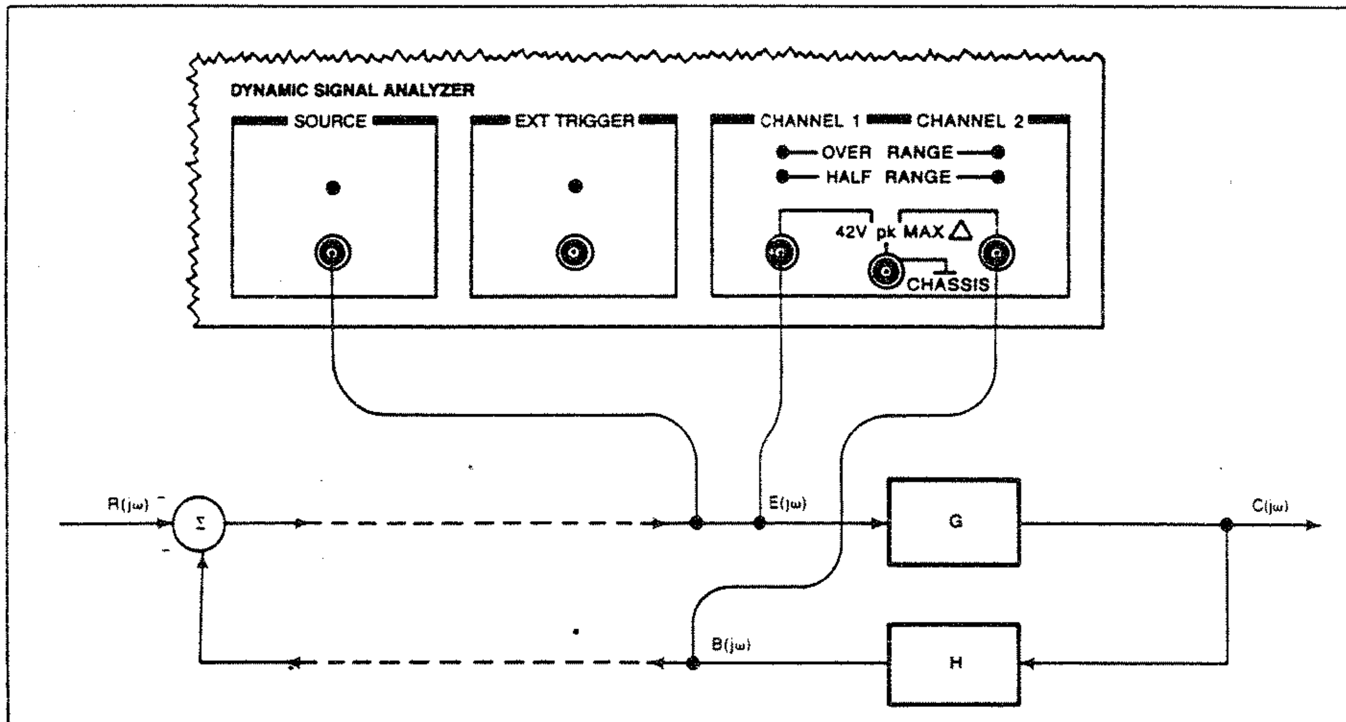


***Fig 1—To take a loop-open direct measurement,** you remove the summing junction from the closed-loop system. You then inject a stimulus signal at E(jω) and record the system's response at B(jω).*

The measurement technique you choose to derive the open-loop response of your closed-loop system depends on the system's stability and nodal accessibility.

testing systems that can operate stably, though possibly at a reduced performance level, with the control loop closed. You can also use this technique to perform maintenance tests on established control systems, as well as to initiate testing of new systems.

When you test a new system, you should reduce its loop gain to a level that's low enough to ensure loop stability. If the tests verify the system design, you can increase the gain to standard operating levels and then retest the system. To perform the loop-closed direct test, you must monitor the signals that enter and exit any summing junction within the main signal path of the control loop. As long as you don't need to test the control loop in the presence of actual operating signals at the reference input, you can use the negative-feedback summing junction.

When you do need to test the control loop in the presence of actual operating signals at the reference input, or when the system's feedback summing junction is inaccessible (for example, when the head-to-disk magnetic interface of a disk drive is the system's feedback summing junction), you must add a summing junction at some other point in the control loop. Note the summing junction added between the two forward blocks, G_1 and G_2, in the closed-loop system in **Fig 2.** (See **box,** "Create summing junctions.")

In the loop-closed direct method, you inject a stimulus signal into the loop at point $S(j\omega)$ and monitor the signals at points $Y(j\omega)$ and $Z(j\omega)$, the inputs and outputs of the summing junction. In this method, the signal at point $Z(j\omega)$ is the reference signal. Assuming that all the energy in the loop derives from the stimulus signal $S(j\omega)$, solving for the signals at $Y(j\omega)$ and $Z(j\omega)$ in terms of $S(j\omega)$ produces the expressions

$$Y(j\omega) = S(j\omega)\frac{G_1G_2H(j\omega)}{1 + G_1G_2H(j\omega)}$$

and

$$Z(j\omega) = S(j\omega)\frac{-1}{1 + G_1G_2H(j\omega)},$$

which is a reasonable assumption to make after you've used averaging to reduce noise. If you take the ratio of $Y(j\omega)/Z(j\omega)$, you obtain the equation

$$Y(j\omega)/Z(j\omega) = -G_1G_2H(j\omega),$$

which is the negative of the open-loop frequency response for the system. (The negation, the result of including the negative feedback in the measurement, represents an additional 180° phase shift in the frequency response.) Some test instruments provide you with either waveform-math capabilities or a calibration constant to remove the additional phase shift.

The loop-closed direct technique is a practical one for measuring the frequency response of any system that can operate, at least minimally, when the loop is closed. In such systems, the loop-closed direct method avoids the problems of changing loading conditions, system saturation, and loss of operation that you encounter when you use the loop-open direct method.

Note, however, that when you use the loop-open direct method, you can't use power-spectrum averaging (the standard form of averaging of a DSA's FFT function) to reduce measurement noise. If you attempt to do so, you'll almost always get the wrong frequency-response measurement. Worse, increasing the number of power-spectrum averages will only reduce the variance of the erroneous result, so the measurement will appear to converge to a valid one.

You can't use power-spectrum averaging because any signals within the system not directly related to the stimulus signal (including system noise and any signals applied to the reference input) would pass through the summing junction unaltered. The signals would, therefore, appear in both channels and maintain constant gain and phase relationships (predictably 0 dB and 0°) from measurement to measurement, making it impossible to average them out. And unless you were to compare your measurement with a known-good measurement, you probably wouldn't notice the error.

In any case, a DSA's signal-processing function removes (from each channel) the nonstimulus-related signals in the loop before the DSA calculates the frequency response. The DSA can be operating either in its SFA (swept-frequency analysis) mode or its FFT mode (using time averaging and a periodic stimulus). If coherent distortion products exist in the nonstimulus-related signals, however, the SFA mode is generally the preferred one.

Loop-closed calculated measurement

The transfer function for the loop-closed calculated measurement technique (**Fig 3**) is $Y(j\omega)/S(j\omega)$. This method resembles the loop-closed direct method except that here the DSA uses the applied stimulus, instead of the signal at $Z(j\omega)$, as the reference signal. With the loop-closed calculated method, however, you can use an FFT's power-spectrum averaging mode (and, there-

If a system's feedback summing junction is inaccessible, you can take response measurements by adding a summing junction at another point in the control loop.

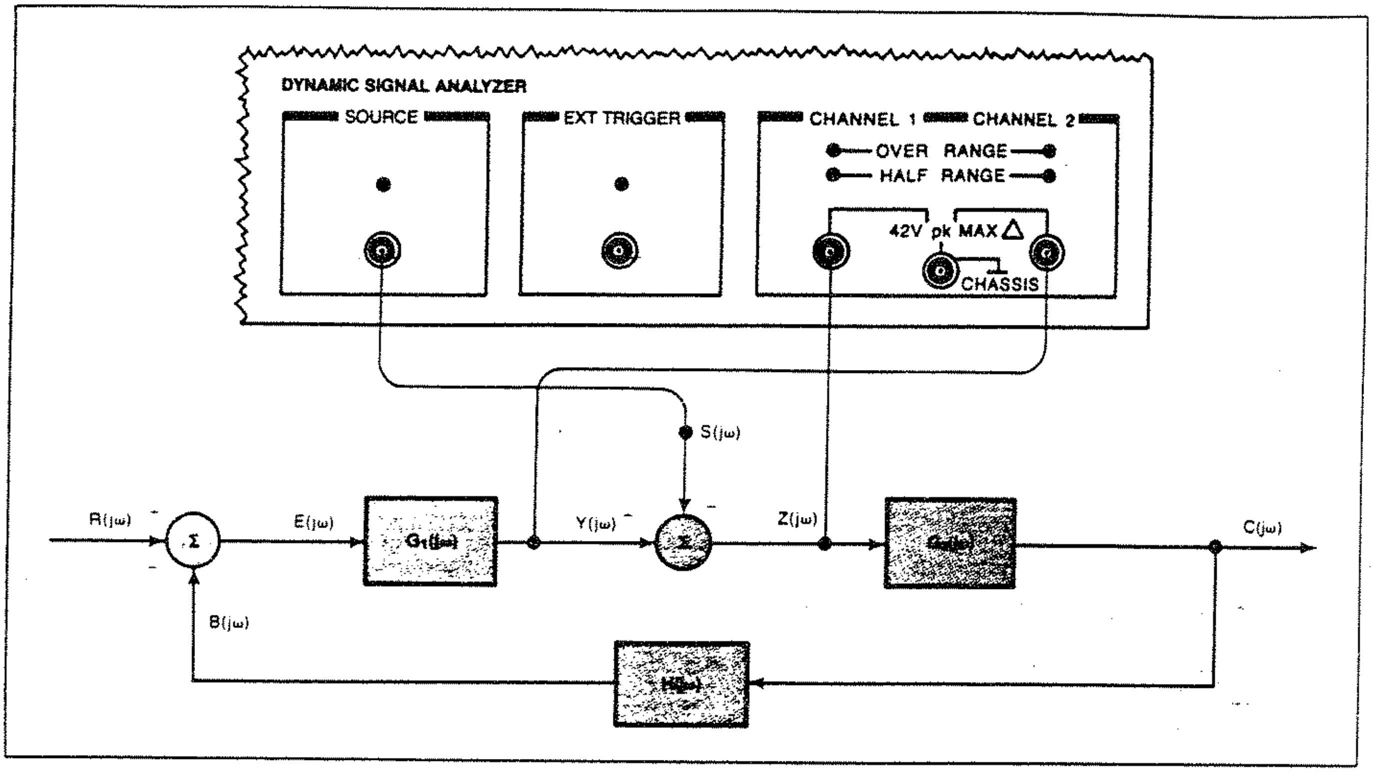


***Fig 2—You can use the loop-closed direct measurement** method only for systems that are stable in the closed-loop configuration. The technique is useful for negative-feedback systems that need periodic maintenance checks.*

fore, you can use random noise) to reject system noise and provide the best linear estimate of a system's frequency response.

You can use random noise because the stray noise in the system appears in only one channel. If you average the relationship between the monitored signals, therefore, eventually the variation (caused by stray noise) in that relationship will drop to zero, as long as the stray noise is not phase-coherent with the stimulus.

The loop-closed calculated technique doesn't limit your selection of an analysis tool. You can use either SFA or FFT analysis with either time or spectrum averaging; for FFT analysis, you can thus use both periodic and nonperiodic stimuli.

The loop-closed calculated method differs from the loop-closed direct method mainly in that, in the loop-closed calculated method, the ratio $Y(j\omega)/S(j\omega)$ does not directly provide the open-loop frequency response, $G_1G_2H(j\omega)$. When you use the equation

$$Y(j\omega) = S(j\omega)\frac{G_1G_2H(j\omega)}{1 + G_1G_2H(j\omega)}$$

to solve for $Y(j\omega)/S(j\omega)$, and then declare the result to be the quantity $T(j\omega)$, you obtain

$$Y(j\omega)/S(j\omega) = \frac{G_1G_2H(j\omega)}{1 + G_1G_2H(j\omega)} = T(j\omega).$$

If you solve for $G_1G_2H(j\omega)$ from this measurement, you obtain

$$G_1G_2H(j\omega) = \frac{T(j\omega)}{1 - T(j\omega)}.$$

Using graphical techniques to perform these calculations would be impractical. DSAs' waveform-math functions, however, perform these calculations automatically. In fact, some of these instruments offer the $T(j\omega)/[1-T(j\omega)]$ calculation as a single-keystroke operation.

The primary disadvantage of the loop-closed calculated technique is that, theoretically, it limits the maximum gain of the open-loop frequency response that you can calculate (from the $Y(j\omega)/S(j\omega)$ measurement) to the dynamic range of one channel of the analyzer. When you measure a signal with a sampling instrument, the instrument's dynamic range is directly

Create summing junctions

To put an electronic summing junction into a control loop, you can use two basic approaches. You can either add new circuitry to the loop to realize the summing junction (**Fig A**), or you can use an existing buffer amplifier (**Fig B**). In the first approach, you can calculate the open-loop frequency response from the measured transfer functions Y(jω)/Z(jω) and Y(jω)/S(jω):

$$\frac{Y(j\omega)}{Z(j\omega)} = -G_1G_2H(j\omega);$$

$$\frac{Y(j\omega)}{S(j\omega)} = \frac{G_1G_2H(j\omega)}{1 + G_1G_2H(j\omega)} = T(j\omega).$$

Therefore,

$$G_1G_2H(j\omega) = \frac{T(j\omega)}{1 - T(j\omega)}.$$

When you're using an existing amplifier (**Fig B**) to make a Y(jω)/S(jω) measurement, you must account for both the gain of the amplifier and the fact that polarities between the Y(jω) and S(jω) legs of the summing junction now match. The following equations account for these factors:

$$\frac{Y(j\omega)}{Z(j\omega)} = -G_1G_2H(j\omega);$$

$$\frac{Y(j\omega)}{S(j\omega)} = \frac{-G_1G_2H(j\omega)}{\frac{R_1}{R_2} + G_1G_2H(j\omega)} = T(j\omega);$$

$$G_1G_2H(j\omega) = \left(\frac{-T(j\omega)}{1 + T(j\omega)}\right)\left(\frac{R_1}{R_2}\right).$$

In all cases, the amplifiers used to implement the summing junction should have bandwidths much greater than the bandwidth of the control system, and they should also have flat frequency responses within the bandwidth of the system.

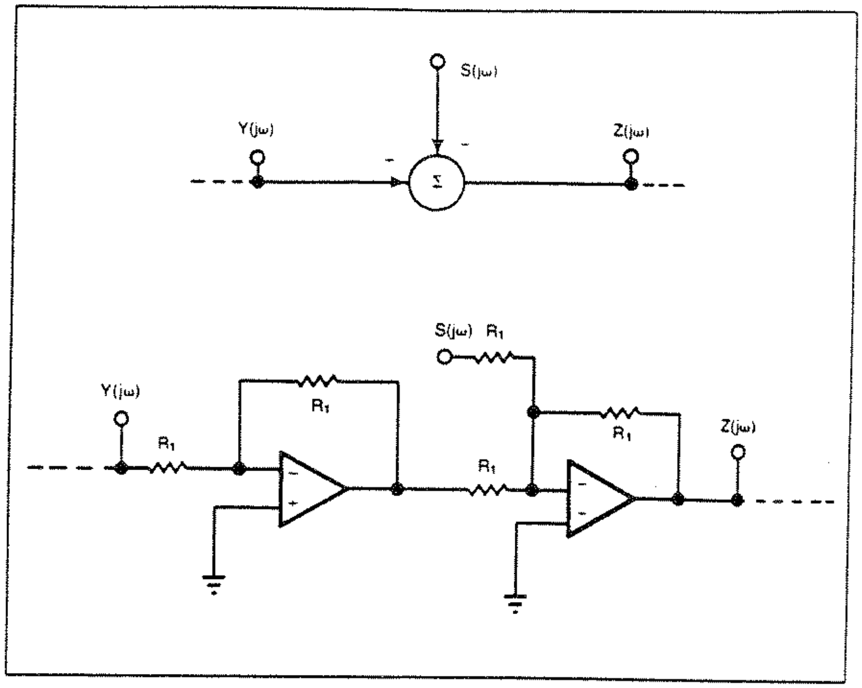


Fig A—This circuit, a typical configuration for adding a summing junction *to a control system, allows you to use the measurements of Y(jω)/Z(jω) and Y(jω)/S(jω) in conjunction with a DSA to calculate the system's open-loop frequency response.*

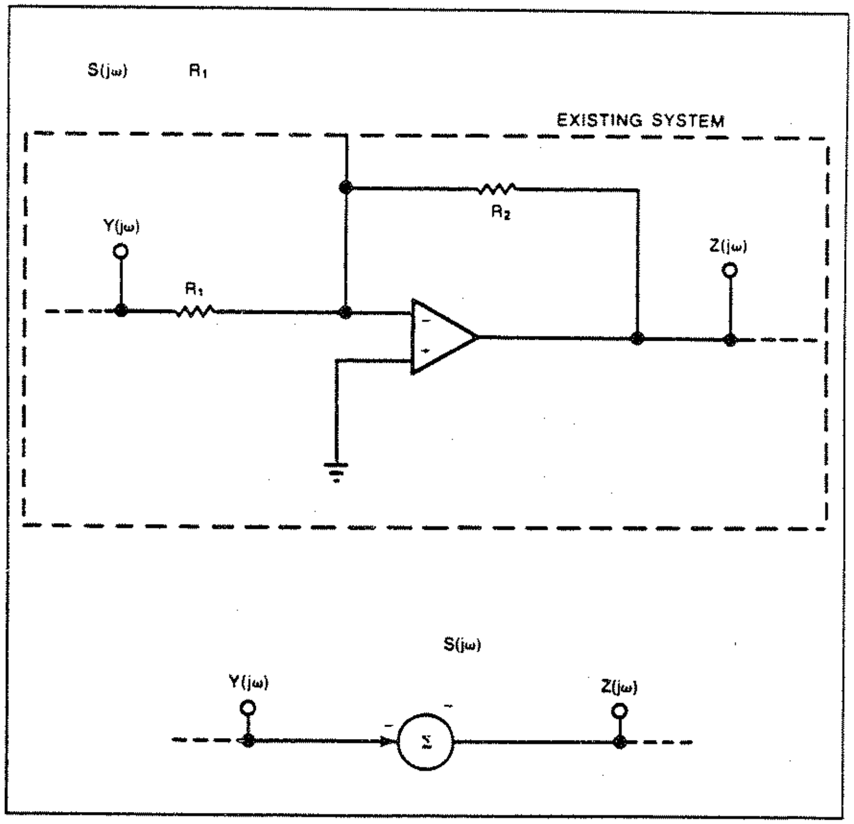


Fig B—You can take advantage of an existing amplifier *in a control loop by simply connecting a resistor to the amplifier's summing junction. Using this technique, you take Y(jω)/Z(jω) and Y(jω)/S(jω) measurements and use the DSA's waveform-math function to calculate the system's open-loop frequency response.*

Loop-closed direct testing is practical for systems that are stable in closed-loop connection, and it avoids the problems of loop-open direct testing.

related to the number of bits of the sampling A/D converter.

In practice, several factors prevent you from reaching even the theoretical limit. For example, if the noise power in the system is much larger than the power of the signal of interest, you must adjust the input sensitivity of the A/D converter to handle the noise. This reduced sensitivity leaves fewer bits of the A/D converter available for resolving the signal of interest.

Mismatch between the input channels of the analyzer can also severely limit the accuracy of the measurement. Unfortunately, the mentioned limitations will not create an obvious distortion of the calculated gain. They may simply produce an unexpected flat region of the open-loop gain in portions of the curve that have very high loop gains.

Design and modeling

Although you develop different control systems differently, you can separate most development into two categories: analytical design and design using frequency-response manipulation. Analytical design is useful for determining the performance of large, expensive systems that must work the first time. In this method, you find the location of the specific poles and zeros of each system component and use frequency-response data and known properties of the system components to help identify those poles and zeros. Once you know the pole/zero locations, you can calculate the system's performance from the models.

Frequency-response manipulation is useful for improving the system's performance. In this technique, you must either alter the components so that their pole/zero locations change or add compensation networks to provide the poles and zeros necessary to generate the required performance. To use this pole/zero-manipulation process, you must work with mathematical representations of the system and obtain accurate models of the system and its components.

When you add compensation networks, however, you don't need to know the exact location and cause of each pole and zero. Instead, you characterize the system completely by its open- and closed-loop frequency responses. You add compensation networks whose frequency responses will constructively change the system's overall frequency response. In effect, this

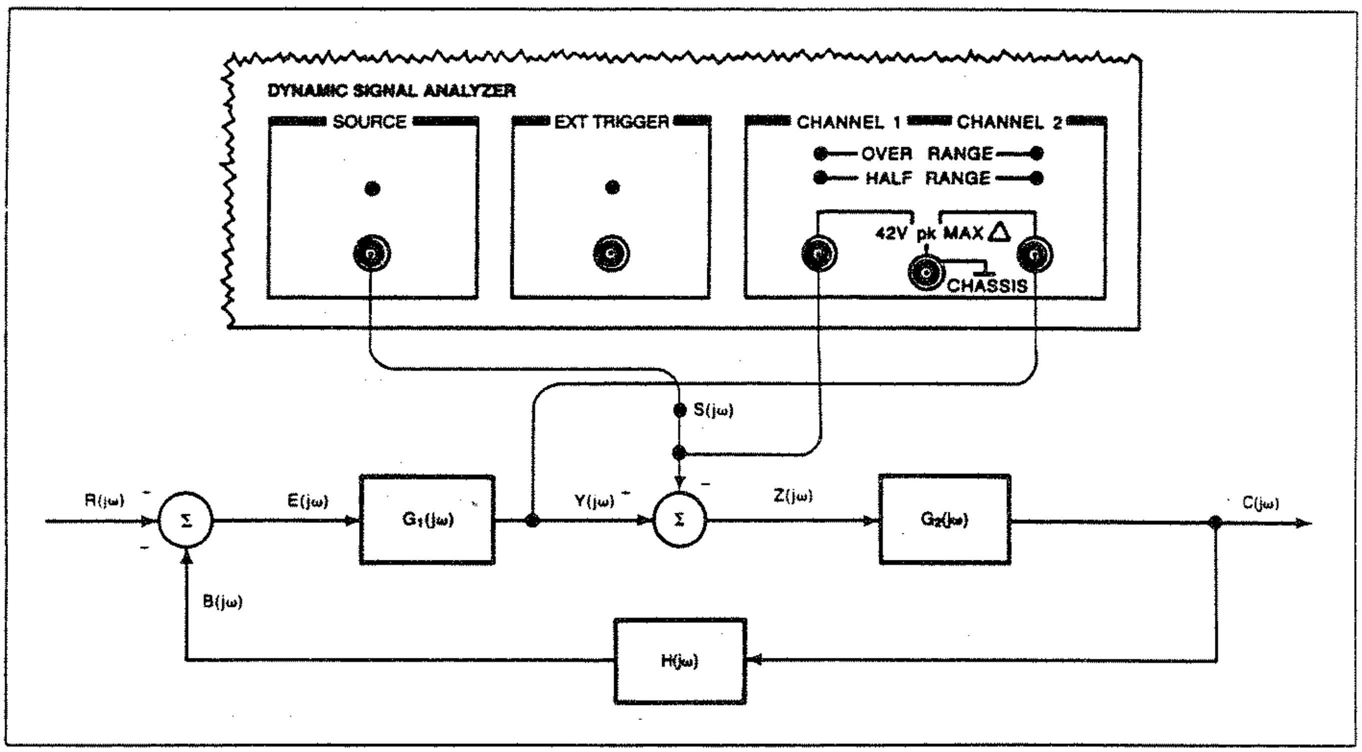


*Fig 3—**Loop-closed calculated measurement** takes advantage of the FFT function's power-spectrum averaging mode. This mode of measurement, when used with a random-noise stimulus, provides the best possible linear estimate of a closed-loop system's frequency response. A DSA's waveform-math capability helps you interpret the results you obtain with this measurement method.*

approach lets you quickly calculate both the frequency response of a compensation network and the network's effect on the system.

The analytical-design and frequency-response-manipulation techniques are similar; the first is simply more concerned with designing with accurate analytical models (typically because of the lack of actual hardware), and the second is more concerned with the measured response of a system.

Typically, you can generate models in two different ways. First, if you have complete knowledge of a device's physical characteristics—such as its mass, friction levels, resistance, and other parameters—you can

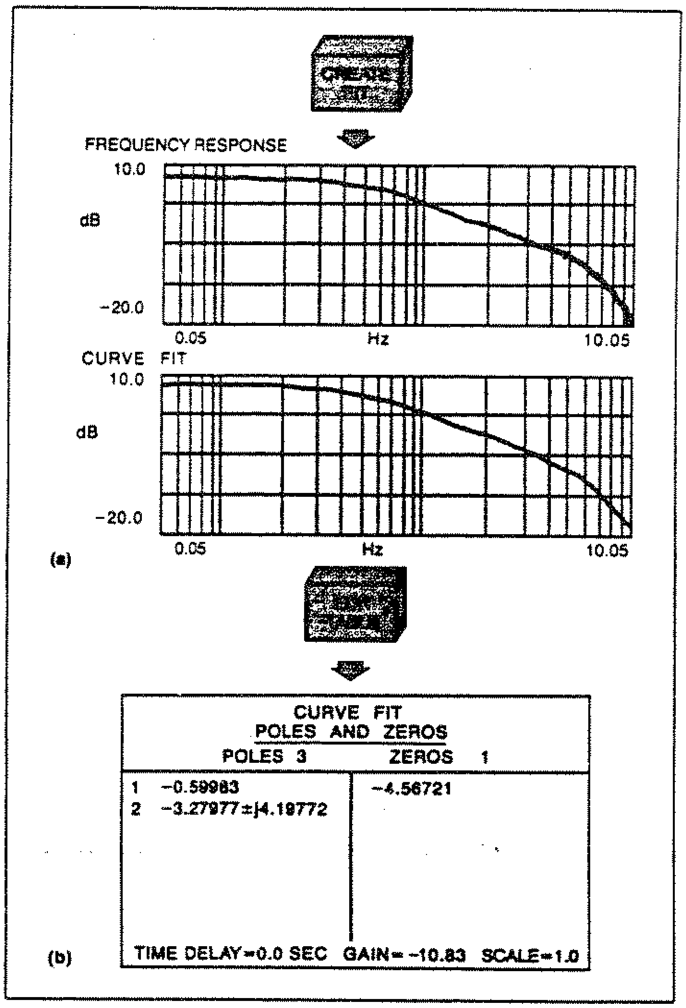


Fig 4—A DSA's curve fitter has an automatic weighting function, which allows the instrument to fit noisy data without producing inconsequential poles and zeros. The DSA displays the system's estimated frequency response below the measured data (a). The edit-table key makes the DSA produce the table of poles and zeros (b) used in the estimate.

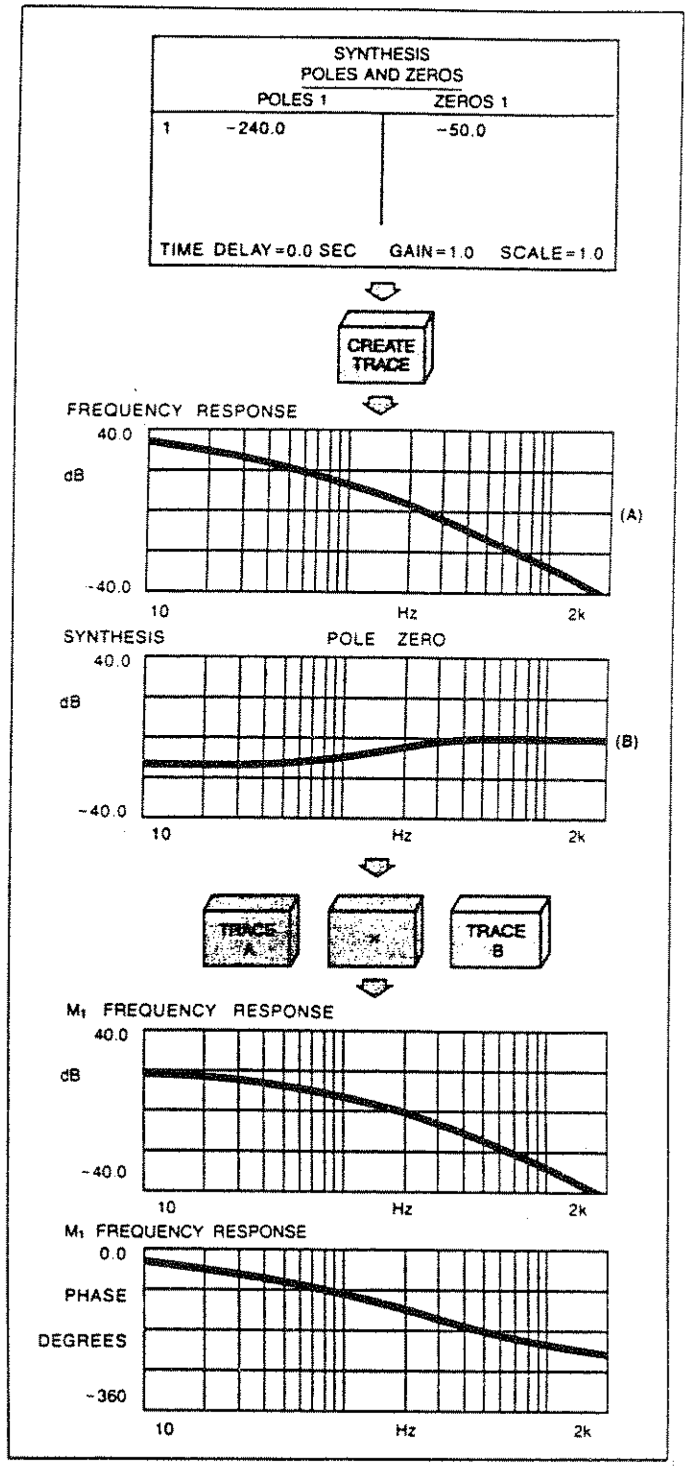


Fig 5—Based on the pole-zero data in a synthesis table, a DSA generates the compensation network's frequency response (trace B). Using waveform math, the DSA multiplies this response by the system's open-loop response (trace A); the results are the gain and phase plots for the compensated system.

In loop-closed calculated measurement, a DSA's FFT function provides a linear estimate of a system's response.

derive a model with them by using Lagrange's equations and other equations. If you don't know the device's physical characteristics, however, you can measure its frequency response and estimate the poles and zeros required to produce that response. When you're developing a system, you'll often have to use both of these techniques.

The most common technique for estimating the poles and zeros required to generate a particular frequency response is to examine a frequency response plotted on a Bode diagram **(Ref 1)** and determine the Q of resonances, the slope at which the system's gain rolls off, and the amount of phase shift through the system in relation to the gain slope. Once you master the technique, you can derive an estimate of the mathematical model for a simple system at a glance.

Using these techniques to estimate the model of a complex system, however, is not easy. For systems with many dominant complex poles, the task of analyzing gain and phase relationships is tedious and cumbersome, and it may result in inaccuracy, because you can't

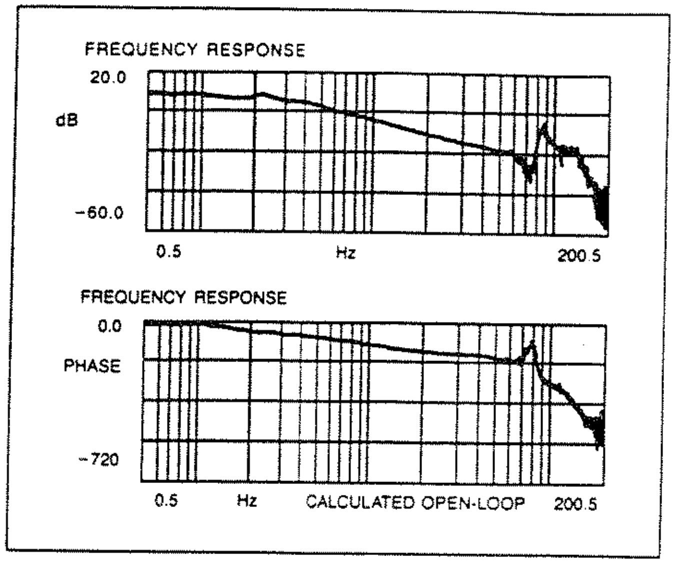


*Fig 8—**A sharp resonance at 90 Hz is evident** in this open-loop frequency response obtained from **Fig 7**'s Y/S measurement. The resonance stems from an excessively long drive shaft between the motor and its load.*

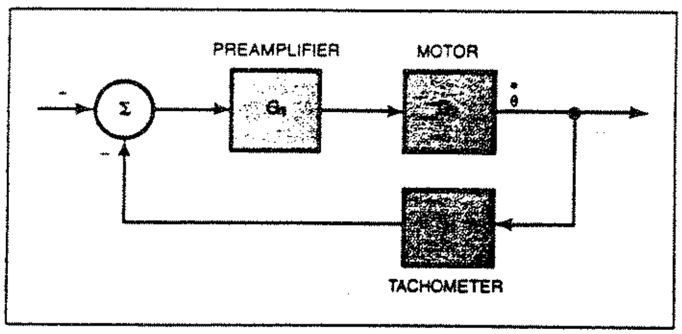


*Fig 6—**This simple motor-speed controller** has all the elements of a negative-feedback, closed-loop system. The system provides an example of how you can exploit a DSA's frequency-response synthesis and waveform-math functions.*

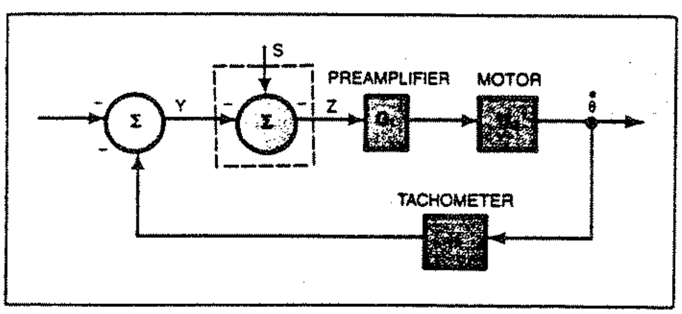


*Fig 7—**To perform operational testing at reduced gain** for the normally unstable speed-control system, you add a summing junction at a point in the system before the preamplifier. This test uses the loop-closed calculated technique (**Fig 3**) to obtain a Y/S transfer function.*

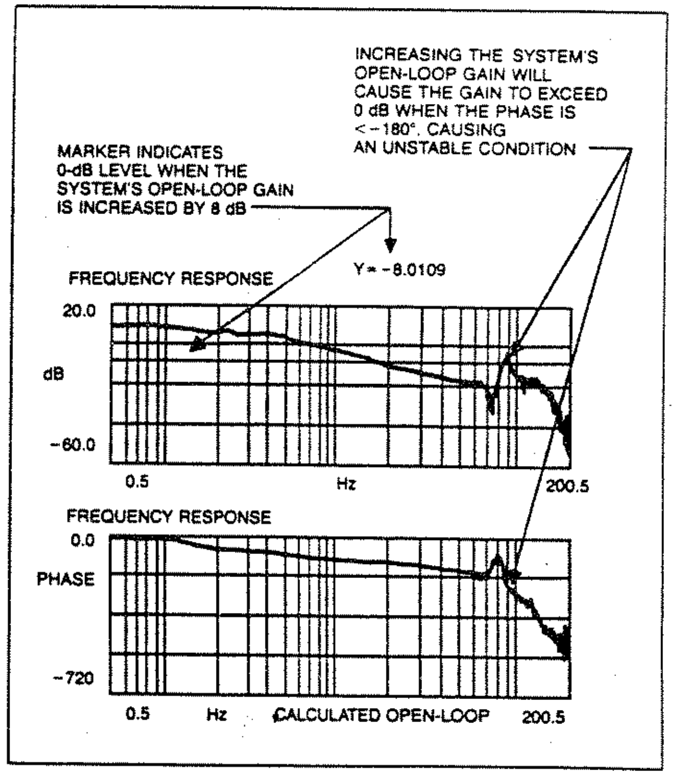


*Fig 9—**When you use the DSA's Y-axis marker to show the system's 0-dB level** during normal operation, you can see that the resonance causes the system to become unstable when gain is adjusted to the desired level. For stability, the system's gain must not exceed 0 dB when phase is more negative than −180°.*

easily sort out the overlapping gain/phase relationships of the many poles and zeros.

Historically, computer programs called curve fitters could automatically analyze a frequency response and produce a pole/zero model. These curve fitters, however, could not distinguish between good data and bad, or noisy, data. If the data was bad, you either had to alter it to remove noise before transferring the data to the computer, or you had to wade through a great deal of inconsequential pole/zero data, which the curve fitter generated to fit the noisy portion of the curve. The computer programs were, therefore, of limited value to designers.

Algorithms called weighting functions improved the usefulness of curve fitters by allowing the computer to analyze the quality of the data before attempting to fit it. Because of these weighting functions, modern curve fitters can successfully process the noisy frequency-response data associated with most control-system measurements, so they can quickly and accurately produce an estimate of a system's pole/zero model.

DSAs contain state-of-the-art curve fitters that are capable of using as many as 40 poles and 40 zeros to estimate the pole/zero model of a frequency response. In a DSA, you can call up the curve fitter with only two keystrokes. When the curve fit is done, the DSA

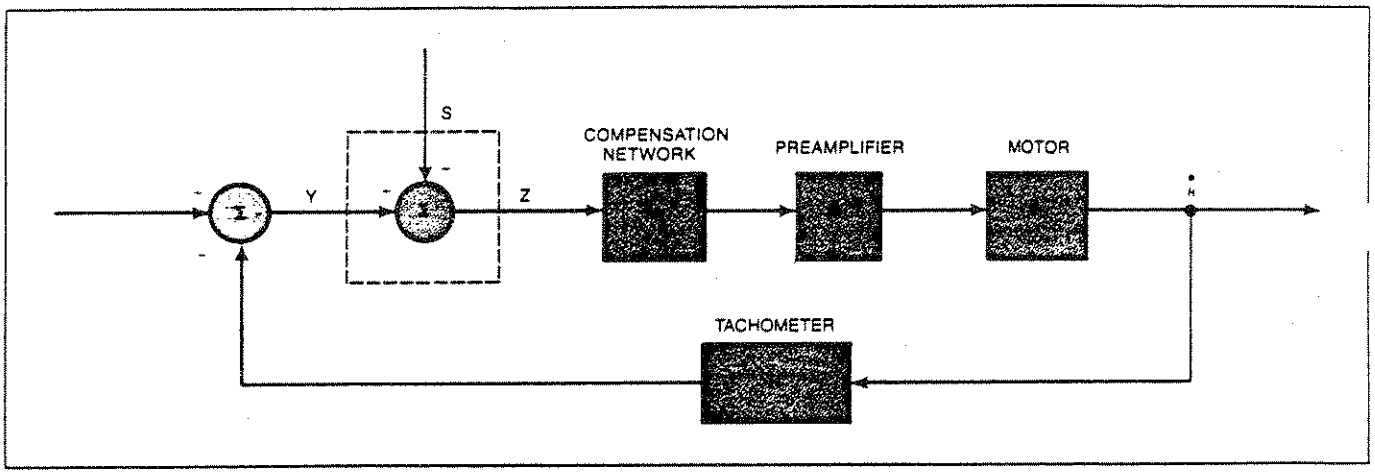


Fig 10—A lowpass filter counteracts the resonance arising from the motor's long drive shaft. Without the filter, the 90-Hz resonance causes instability when the system operates at its normal gain setting.

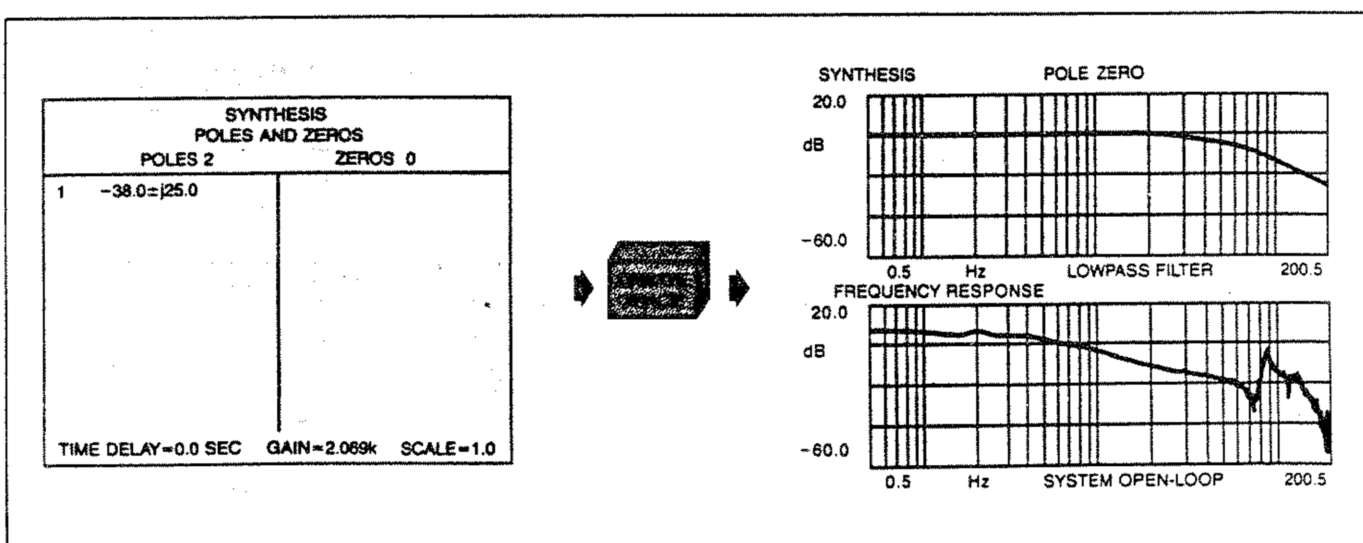


Fig 11—The DSA's frequency-response-synthesis function uses pole-zero data and waveform math to create the lowpass filter's response curve, which is displayed above the speed-control system's frequency response.

A DSA simplifies the task of determining the proper compensation network needed to stabilize closed-loop control systems.

calculates a frequency response from the model and displays that frequency response below the measured frequency response for comparison (**Fig 4a**). Another keystroke displays the table of poles, zeros, and gain (**Fig 4b**) that the curve fit produces.

After you estimate the model of a system or a compensation network, you must calculate the model's frequency response. If you do this calculation manually, you should use Bode's graphical techniques. If you use a DSA, however, you can obtain a higher degree of accuracy for both simple and complex systems.

A DSA's frequency-response-synthesis program performs the inverse function of an advanced dedicated curve fitter, allowing you to enter transfer functions that have as many as 40 poles and 40 zeros into the analyzer. The DSA then calculates and displays the frequency response. The programs also allow you to enter gain and delay parameters.

The frequency-response-synthesis function lets you immediately assess the effect a cascade compensation network will have on your system. By displaying the measured frequency response of a system on one trace and the synthesized frequency response of a compensation network on another, the DSA lets you make either a quick visual evaluation or a precise calculation of the combined frequency response of the two (**Fig 5**). (To make a precise calculation of the responses, you multiply the two frequency responses using the DSA's waveform-math function).

Because curve fitters generate only linear models, the result of the curve fit will be a linear approximation of the device's operation. Should a nonlinearity (other than random noise) produce a large number of poles and zeros, you can reduce the order of the model by simply transferring the curve-fit data to the DSA's frequency-response-synthesis function and selectively subtracting or adding poles and zeros to obtain the best model (with the fewest poles and zeros).

You can then quickly assess the effects of your modifications by synthesizing the frequency response of the new model and comparing it with the measured response. The initial estimate provided by the curve fitter and the modifications handled by the frequency-response-synthesis function greatly reduce the time required to produce an adequate linear representation.

A case study

The system in **Fig 6** provides an example of how you can use frequency-response synthesis and waveform math to develop a relatively simple motor-speed controller. We constructed a prototype of the system and found, upon power-up, that the motor's speed was unstable.

Reducing the gain of the preamplifier by 8 dB stabi-

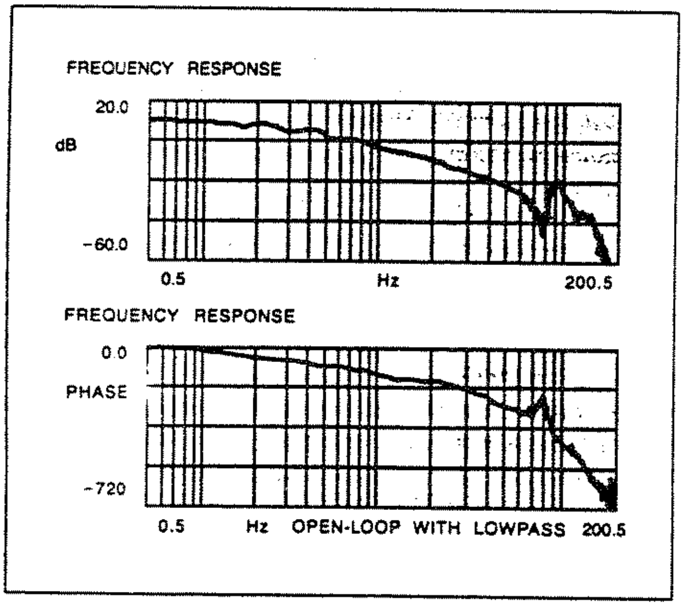


*Fig 12—**To obtain these curves,** which are products of the responses of the system and the lowpass filter, you use the DSA's waveform-math function to perform the multiplication. The curves indicate that the use of the filter stabilizes the speed-control system.*

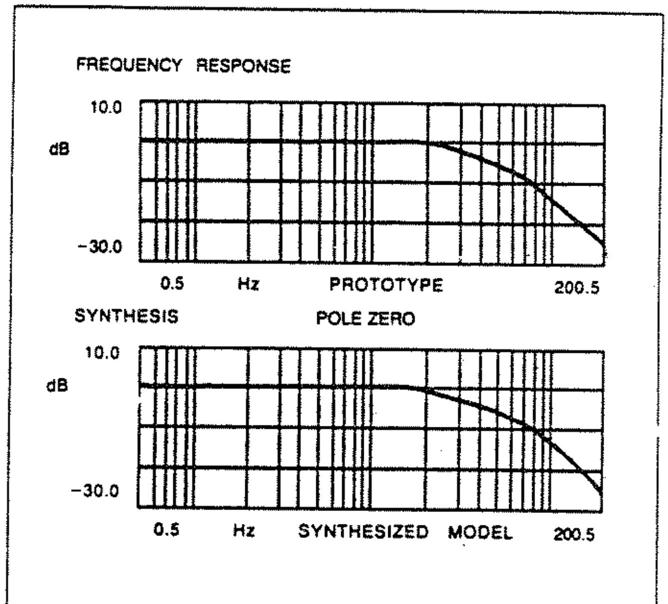


*Fig 13—**To check the design** of the lowpass filter, you can compare the filter's measured (upper trace) and theoretical (lower trace) frequency responses.*

The curve-fitter function in a DSA automatically analyzes a system's frequency response and produces a pole/zero plot.

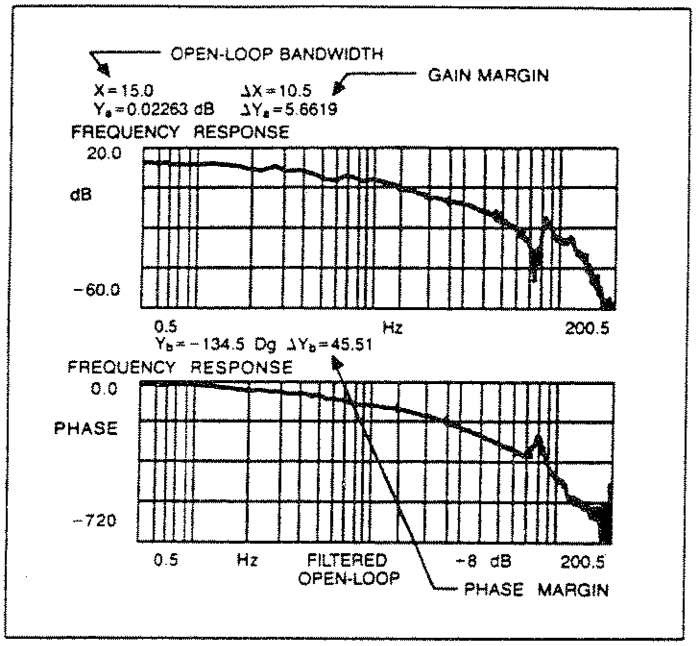


***Fig 14—The frequency response of the compensated speed-control system** shows that the lowpass-filter compensation network completely stabilizes the system. The DSA gives direct readouts of gain margin and phase margin.*

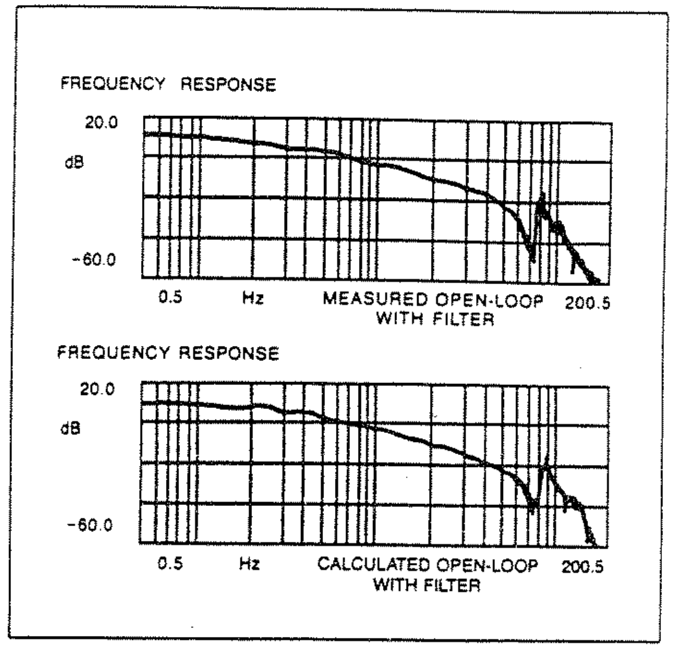


***Fig 15—A final check on the speed-control system** shows that the system's measured frequency response (upper trace) closely matches the response predicted (lower trace) by the DSA's frequency-response synthesis and waveform-math function.*

lized the system, but degraded its performance below acceptable levels. While the system was at the reduced-gain level, we measured the open-loop frequency response by adding a summing junction to the system just before the preamplifier (**Fig 7**) and by using the loop-closed calculated (Y/S) measurement technique. **Fig 8** shows the open-loop frequency response calculated from the Y/S measurement.

The measurement revealed a sharp resonance at approximately 90 Hz. When we placed the Y-axis marker at −8 dB to indicate the unity-gain (0-dB) location during normal system operation (**Fig 9**), we saw that the resonance did indeed cause the system to become unstable when gain was adjusted to the desired level.

The source of the resonance was a relatively long drive shaft attached between the motor and the anticipated system load. Redesigning the system would be too expensive, so we decided not to alter the drive shaft, but to find an electronic solution.

Because the problem was occurring at a relatively high frequency, we added compensation to the system in the form of a lowpass filter (**Fig 10**). Using the DSA's frequency-response-synthesis function, we entered the pole locations for a simple lowpass filter into the analyzer and synthesized and displayed the frequency response above the system's measured frequency response (**Fig 11**).

By using its waveform-math function to multiply the two frequency responses, the DSA predicted the effect of adding the lowpass filter to the system's loop. **Fig 12** shows the gain and phase of the modified system. The predicted response of the modified system indicated that the lowpass filter would be a good solution to the resonance problem, so we constructed a prototype of the filter.

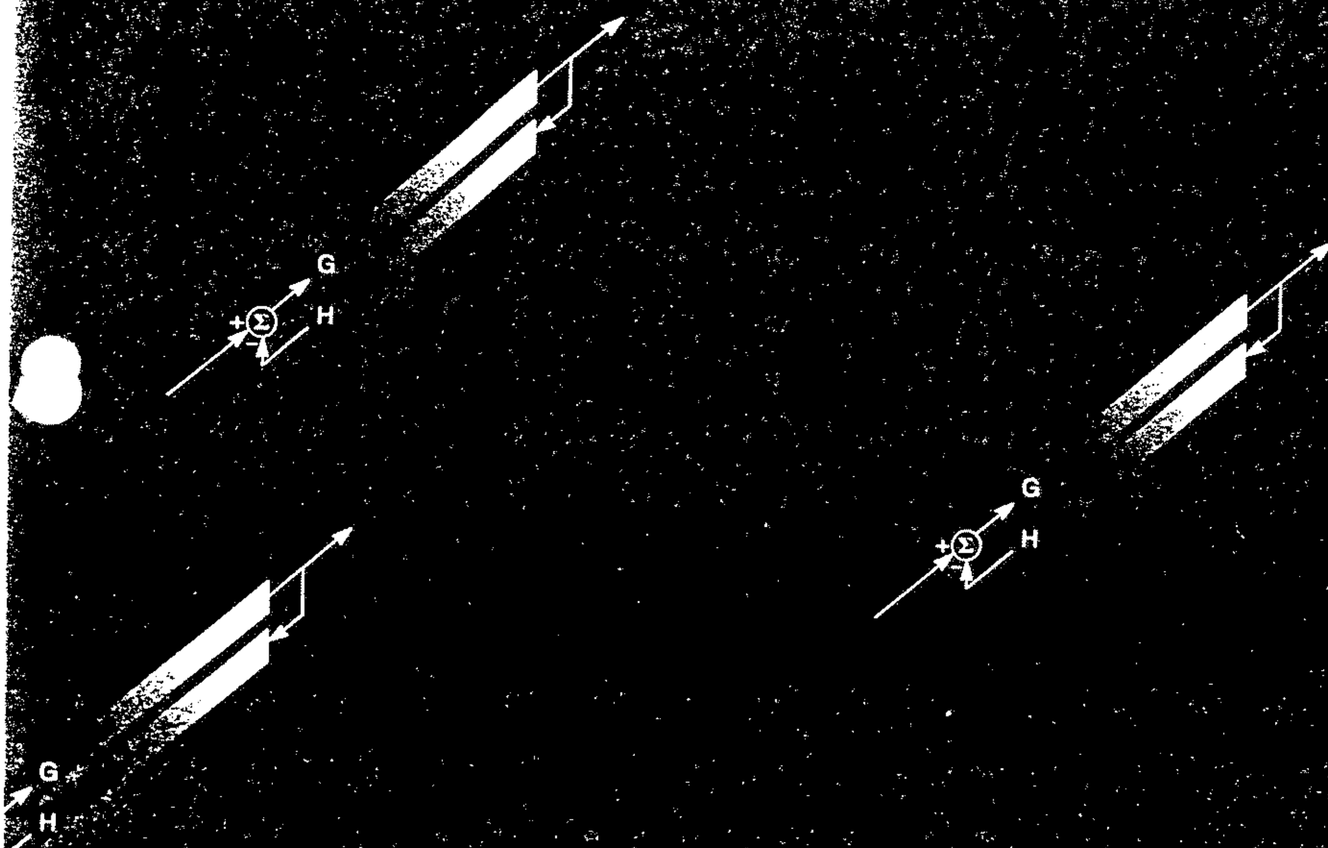
To ensure that the design was correct, we measured and compared the frequency response of the filter prototype to its synthesized frequency response (**Fig 13**). Once we had confirmed its design, we added the filter to the system and increased the preamplifier's gain by 8 dB to its previous level. The system remained stable, but we measured the open-loop frequency response again to make sure that the gain and phase margins were sufficient to maintain stability.

The measurement indicated that the resonance had indeed been attenuated far below any level of concern and, using the analyzer's markers, we quickly recorded the gain margin, phase margin, and open-loop bandwidth (**Fig 14**). To evaluate the integrity of the design approach, we compared the measured open-loop frequency response of the modified system with the pre-

HEWLETT PACKARD

CONTROL SYSTEM DEVELOPMENT USING DYNAMIC SIGNAL ANALYZERS

APPLICATION NOTE 243-2



Control System Development Using Dynamic Signal Analyzers

Dynamic Signal Analyzers (DSAs) represent a new generation of microprocessor-based test instruments designed to support the development of control systems. By combining the computational resources of microprocessors with the accuracy of precision measurement hardware, DSAs combine high-performance measurements and powerful computer-aided-engineering. By consolidating this much power into a single instrument, DSAs have expanded the role of test instruments beyond traditional testing functions to include contributions in the areas of modeling, design and analysis.

The purpose of this application note is to examine how the advanced measurement and analysis capabilities of a DSA can be applied to the development and production of control systems to reduce testing time, reduce analysis time, provide more information from measurements and, in general, enhance the overall development and production process.

Using This Application Note

This application note is designed for both the experienced control systems engineer who may be unfamiliar with DSAs and the experienced DSA user entering the field of control systems. To accommodate this broad range of readers, the note is divided into two parts.

Part 1 is a review of the basic concepts associated with control systems and linear control theory. This section serves as a general resource and may be considered optional reading for the experienced control system engineer.

Part 2 is an introduction to the features and functions of DSA's which directly contribute to the development of control systems. Each feature or function is briefly described with example applications provided.

A glossary of control system terms is provided in Appendix A.

Contents

Part I

Part 1: **An Introduction to Control Systems and Classical Control Theory.**

Chapter 1: Basic Terms and Definitions

A control system has been formally described as, "A system in which deliberate guidance or manipulation is used to achieve a prescribed value of a variable."[1] With a variable further defined as, "a quantity or condition which is subject to change," it becomes apparent that the components of a control system may be virtually any definable entity, be it electrical, mechanical, biological, organizational or otherwise.

The human circulatory system, pacemakers, motor speed controls, clothes dryers, automobile cruise controls and voltage regulators are a few examples of the vast number of control systems in existence. The diversity of control systems may at first seem a barrier against the development of a common analysis and design strategy. Fortunately, if the components of a system can be represented through a common mathematical symbolism, then there exists a collection of concepts and methods for studying the physical properties of control systems known as *control theory.*

While a thorough study of control theory is far beyond the scope of this application note, the following paragraphs present the basic concepts associated with classic control theory as applied to continuous linear control systems.[2]

To categorize control systems with common traits or functions, several subclasses of control systems have been defined. One of the basic categories of control systems are those systems which operate without human intervention. Control systems in this category are called *automatic control systems.* An example of an automatic control system is an automobile cruise control which maintains the speed of the vehicle without attention from the driver. If the driver disengages the cruise control, he then becomes part of the control system regulating the speed of the car and, therefore, part of a nonautomatic control system.

Another category involves those automatic control systems which involve mechanical motion as the controlled variable. These control systems are called *servomechanisms* (commonly referred to as *servos*) and are defined as, "An automatic feedback control system in which the controlled variable is mechanical position or any of its time derivatives." While this definition seems straightforward, general usage has diluted the literal meaning to include virtually any electronic, electro-mechanical or mechanical control system.

Control systems are also categorized as being either *open-loop* or *closed-loop.* The difference between these two categories, the use of feedback, becomes easier to understand when viewing the basic model of a control system. Formal definitions of open-loop and closed-loop control systems have therefore been incorporated into the following chapter on control system modeling.

Chapter 2: Modeling

The first step in the design or analysis of a control system is to develop an analytical model of the system. This is done by dividing the control system into functional blocks. Each block may represent any portion of the control system from an individual component to a group of components which perform an identifiable function.

[1] American National Standards Institute specification MC85.1M-1981, *Terminology for Automatic Control.*

[2] References for further study of modern or classic control theory as applied to linear, nonlinear, continuous and discrete control systems are listed at the end of this note.

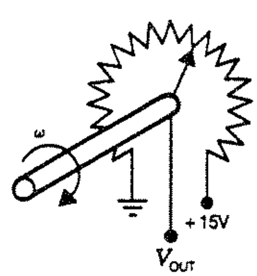
2-1: The Open-Loop Model

Figure 1-1a is a block diagram which represents a very basic control system. The letters r and c represent the directly controlled variable and the reference input respectively. The letter g represents an equation which describes the influence of the elements within the functional block on a signal or action compared at the input and output of the functional block. All lower case letters generally denote functions in the time domain unless otherwise specified (for example, $c = c(t)$). The upper case variables R and C in Figure 1-1a represent the Laplace transform of r and c expressed as functions of the complex variable s[1]. The upper case G represents the Laplace transform of g and is generally referred to as a *transfer function*[2].

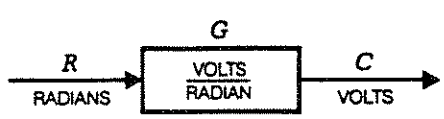
A simple example of the type of control system shown in Figure 1-1a is a potentiometer connected as a voltage divider, as shown in Figure 1-1b. For this example the reference input R would have units of radians, the directly controlled variable C units of volts, and the transfer function G would be a constant with units of volts per radian (as shown in Figure 1-1c). A drawback of this type of a control system is its inability to respond to dynamic changes in the system. For example, if a load resistance was connected to the output, there would be an undesirable change in the output voltage. This type of control system, which cannot take corrective action to alleviate undesirable changes of the directly controlled variable, is called an *open-loop control system.*

FIGURE 1-1.

a. Control System Block Diagram (Open-Loop).



b. Control System Corresponding to Block Diagram of Figure 1a.



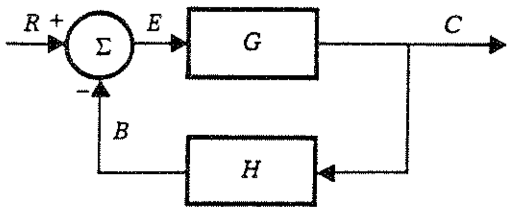
R - FUNCTION WITH UNITS OF RADIANS.
G - TRANSFER FUNCTION WITH UNITS OF VOLTS/RADIAN.
C - FUNCTION WITH UNITS OF VOLTS.

c. Detailed Block Diagram of Control System Shown in Figure 1b.

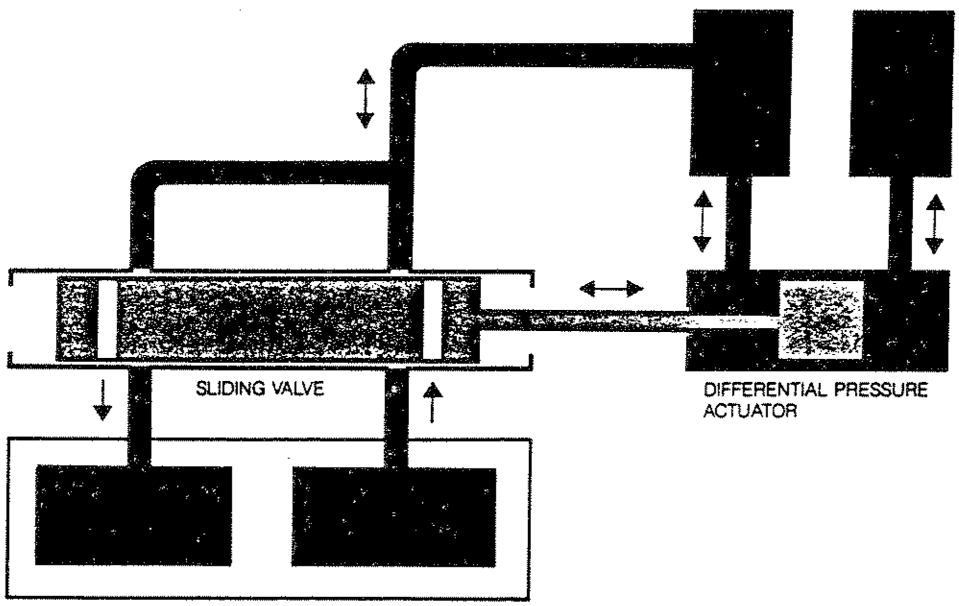
[1] In general, capital letters denote transformed quantities. The quantities may be either Laplace transformed as a function of the complex variable s, (e.g., $G(s)$), or Fourier transformed as a function of the frequency variable $j\omega$, (e.g., $G(j\omega)$). Functions of s are generally abbreviated to their capital letter only (i.e., $G(s)$ is abbreviated to G). Functions of $j\omega$, however, are never abbreviated.

[2] A transfer function is defined as the ratio of the Laplace transform of the output to the Laplace transform of the input in the absence of all other signals, and with all initial conditions zero. Input and output refer to the signals or variables applied to and delivered from a system or element, respectively.

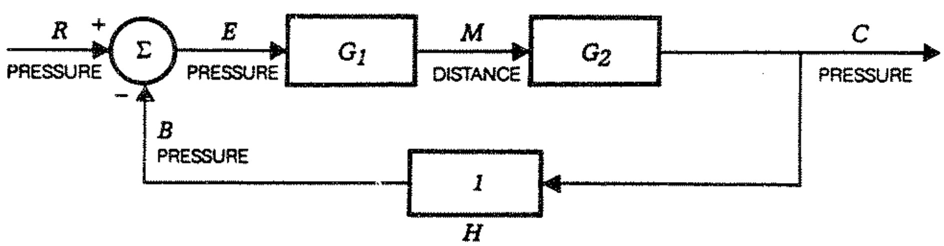
FIGURE 1-2.



a. Control System Block Diagram (Closed-Loop).



b. Control System Corresponding to Block Diagram of Figure 1-2.a.



G_1 = FUNCTION WITH UNITS OF $\frac{\text{DISTANCE}}{\text{PRESSURE}}$ G_2 = FUNCTION WITH UNITS OF $\frac{\text{PRESSURE}}{\text{DISTANCE}}$

c. Detailed Block Diagram of Control System Shown in Figure 1-2.b.

2-2: The Closed-Loop Model

Another basic form of control system is shown in Figure 1-2a. In this system the output C is fed back through a functional block with a feedback transfer function H and compared to the reference signal R via a summing junction. The signal resulting from the difference between R and the feedback signal B is called the *error* or *actuating* signal E. The principal advantage of this form of system is that any change in C, with R remaining constant, causes a change in E, ($E = R-B = R-CH$). If the system is operating properly, the change in E forces C to return to the point where the value of B approaches the value of R. The effect is that the output is maintained at a desired value despite disturbances to the system. This type of control system is called a *closed-loop control system* and is defined as any control system in which the directly controlled variable has an effect upon the input quantity in such a manner as to maintain the desired output level.

An example of this second form of control system is illustrated by the pressure regulator shown in Figure 1-2b. The objective of this system is to adjust the pressure in Tank 2 (P_2) until it is equal to the pressure in Tank 1 (P_1).

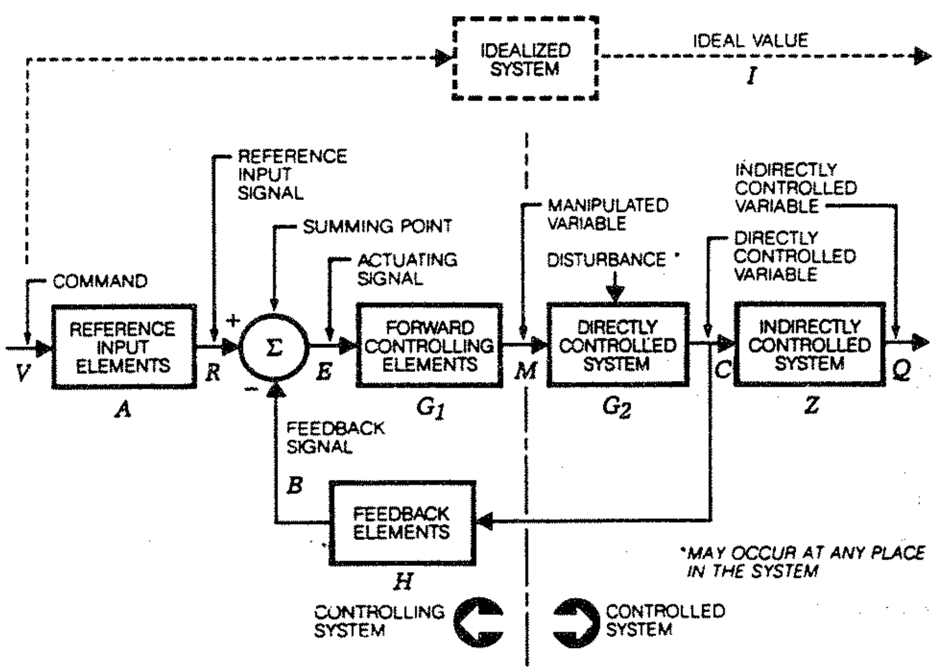
Figure 1-2c is one possible block diagram for this system. In this block diagram the function of the differential pressure actuator is represented by a summing junction and a forward transfer function G_1. The action of the sliding valve is then represented by the forward transfer function G_2. A perfectly valid alternative would be to combine G_1 and G_2 into a single forward transfer function G. The resultant block diagram would then have the same form as Figure 1-2a.

This control system is also an example of a system in which the controlled variable is fed back to the summing junction without any modification; the transfer function, H, is simply equal to 1. This type of control system is called a *unity feedback control system*.

A general block diagram illustrating most of the elements of an automatic closed-loop control system is shown in Figure 1-3[1].

FIGURE 1-3.

BLOCK DIAGRAM OF AUTOMATIC CONTROL SYSTEM



[1] Figure 1-3 is adapted from the American National Standard ANSI MC85.1M-1981, *Terminology for Automatic Control.*

The primary objective in designing a control system is to construct a system that achieves the desired output level as fast as possible and maintains that output with little or no variation. One of the first techniques developed to measure a control system's compliance with these design goals was the step response.

3-1: Time Domain Performance

Step Response

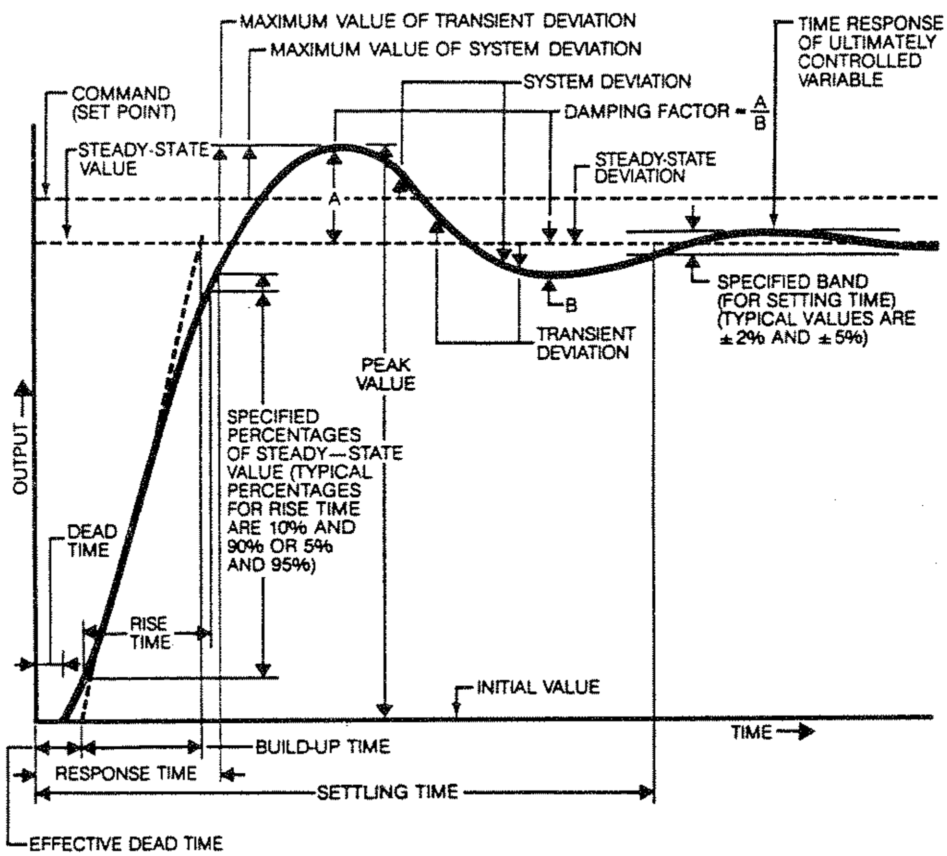
The step response is the measured reaction of the control system to a step change in the input. A typical step response and its associated parameters are illustrated in Figure 1-4[1].

The step response has several favorable characteristics which have maintained its universal acceptance and popularity:

- the step stimulus is easy to generate
- the stimulus is easily modeled [$u(t)$] making the solution to the differential equation (used to predict the system's time domain response) much less complicated
- several measurement techniques are available for recording the time domain response to the step input
- key aspects of the control system's performance can be derived from the step response.

FIGURE 1-4.

TYPICAL TIME RESPONSE OF A SYSTEM TO A STEP INCREASE OF INPUT



[1] Figure 1-4 is adapted from the American National Standard ANSI MC85.1M-1981 *Terminology for Automatic Control.*

There are several measures of performance which can be derived from the step response. The rise time of the step response provides a measure of how fast a system can initially achieve the desired output level. The maximum overshoot (shown in Figure 1-4 in terms of either peak value or maximum value of transient deviation) provides a relative measure of the maximum output level resulting from a specific input. The steady-state deviation indicates a constant error in achieving a desired output. Settling time, perhaps the most significant parameter, is a measure of how long it takes the system to settle to its steady-state value.

If the system never settles to its steady-state value (for example, it constantly oscillates about a desired output), the system is considered *unstable.* Taken one step further, the settling time can be interpreted as a relative measure of stability, with a short settling time considered more stable than a long settling time.

In addition to the step response, there were two other early stimulus signals: the ramp function [$tu(t)$] and the parabolic function [$t^2u(t)$]. These signals provided the same simplicity in modeling as the step response and also provided a means of measuring a control systems ability to track dynamic signals.

3-2: Frequency Domain Performance

The time domain responses to the step, ramp and parabolic forcing functions were the only universally accepted techniques for measuring the performance of a control system until the early 1930s. It was during this period that three Bell Laboratories scientists, H.S. Black, H.W. Bode and H. Nyquist, were doing pioneering work on the characterization of control systems in the frequency domain. In an attempt to provide amplifiers with better linearity, Black began a rigorous study of the effects of negative feedback on electronic amplifiers (a basic form of automatic closed-loop control system). Early experiments resulted in several observations including improved linearity and, in some cases, unexpected oscillations in the amplifier's output. It was the unexpected oscillations which inspired Nyquist to study the cause of such instabilities in closed-loop control systems. From his studies, Nyquist discovered that the stability of a closed-loop system could be determined from a simple frequency response plot. Before discussing Nyquist's discovery, it is helpful to review a few of the basic definitions and concepts associated with the frequency domain aspects of a control system.

3-2.1: Frequency Domain Terms and Definitions

One of the most important transfer functions associated with a closed-loop control system relates the directly controlled variable C to the reference input. The ratio C/R is referred to as either the *control ratio* or the *closed-loop transfer function*; this note refers to it as the latter. By solving for C/R in terms of G and H we have: $C/R = G/(1+GH)$, as shown in Figure 1-5. As previously mentioned, capital letters with no subscripts represent transformed quantities expressed as a function of s. The closed-loop transfer function can therefore be expressed as:

$$\frac{C(s)}{R(s)} = \frac{G(s)}{1 + G(s)H(s)} = \frac{G(s)}{1 + GH(s)}$$

Important values of s are those values which set the numerator and/or denominator of the closed-loop transfer function equal to zero. Values of s which set the numerator to zero are called zeros of the closed-loop transfer function or *closed-loop zeros.* Values of s which set the denominator equal to zero (i.e., s such that $1 + GH(s) = 0$) are called poles of the closed-loop transfer function or *closed-loop poles.*

At this point it is important to note that the complex variable s can be further expressed in terms of the variables σ and $j\omega$. That is, $s = \sigma + j\omega$ where σ represents the real or damping component of s, and $j\omega$ represents the imaginary or frequency component of s.

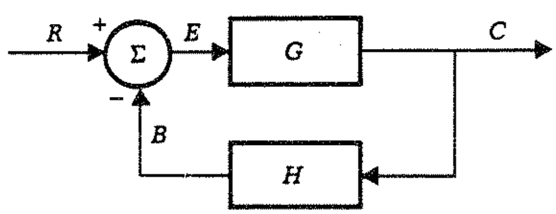
A common tool used to study control systems is a graph called the *s-plane*. The *s*-plane is a two-dimensional Cartesian graph which represents values of s. The ordinate of the s-plane represents the imaginary part of s (i.e., $j\omega$), and the abscissa represents the real part of s (i.e., σ). If values of s which constitute the closed-loop poles are plotted with X's on the s-plane and the values which constitute closed-loop zeros are plotted with O's, the result is a pole/zero plot of the closed-loop transfer function as shown in Figure 1-6.

When the magnitude of the closed-loop transfer function is plotted as a third axis of the s-plane, the effects of the poles and zeros on the magnitude of the closed-loop transfer function at any value of s can be quickly realized as shown in Figure 1-7.

Figure 1-7 shows only the left half of the s-plane to illustrate the contour of $|C/R|$ for values of s along the $j\omega$ axis (i.e., for values of s equal to $0 + j\omega$). This contour is significant in that it represents the same curve produced by evaluating the magnitude of the Fourier transform of c divided by the Fourier transform of r for positive values of (i.e., $|C(j\omega)/R(j\omega)|$ for values of $\omega \geq 0$). Therefore, this contour also represents the gain-versus-frequency plot obtained by physically measuring the gain of a control system between its input and output.

FIGURE 1-5.

SOLVING FOR $\frac{C}{R}$ IN TERMS OF G AND H



$$C = EG$$
$$E = R - B$$
$$B = CH$$

SOLVING FOR C IN TERMS OF C, G, H AND R WE HAVE:

$$C = RG - BG$$
$$C = RG - CHG$$

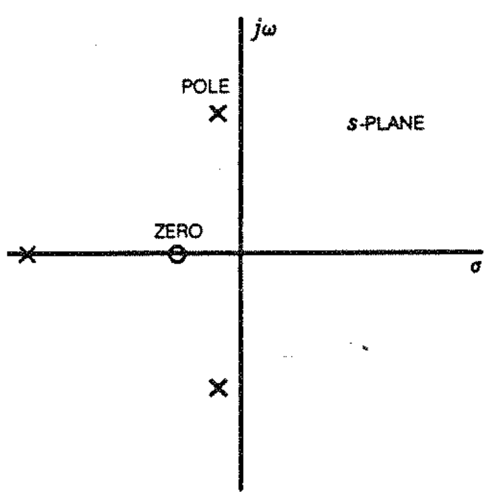
SOLVING FOR $\frac{C}{R}$ WE HAVE:

$$C = \frac{RG}{1 + GH}$$

$$\frac{C}{R} = \frac{G}{1 + GH}$$

FIGURE 1-6.

POLE/ZERO PLOT OF THE CLOSED LOOP TRANSFER FUNCTION $\frac{G(s)}{1 + G(s)H(s)}$



A similar diagram can be drawn for the phase of $C(s)/R(s)$ as shown in Figure 1-8. Again the contour presented by the values of $\angle C(s)/R(s)$ along the $s = 0 + j\omega$ axis represent $\angle C(j\omega)/R(j\omega)$ for positive values of ω. This contour also represents the phase-versus-frequency plot obtained by physically measuring the phase shift of a control system between its input and output.

The information provided by the highlighted contours in Figures 1-7 and 1-8 represents the frequency-dependent relation between steady-state sinusoidal input signals ($R(j\omega)$) and the resulting steady-state sinusoidal output signals ($C(j\omega)$), that is, they represent the *frequency response* of the device characterized by C/R.

For transfer functions in general, the information produced by evaluating the Fourier transform for all values of $j\omega$ can be regarded as a subset of the overall contour produced by evaluating the Laplace transform for all values of s. The Fourier transform of a transfer function evaluated for positive values of ω also represents the physically measured gain and phase relationship (i.e., frequency response) between the input and output of the device modeled by the transfer function.

FIGURE 1-7.

MAGNITUDE PLOT OF $\dfrac{G(s)}{1 + GH(s)}$ **VERSUS VALUES OF** s

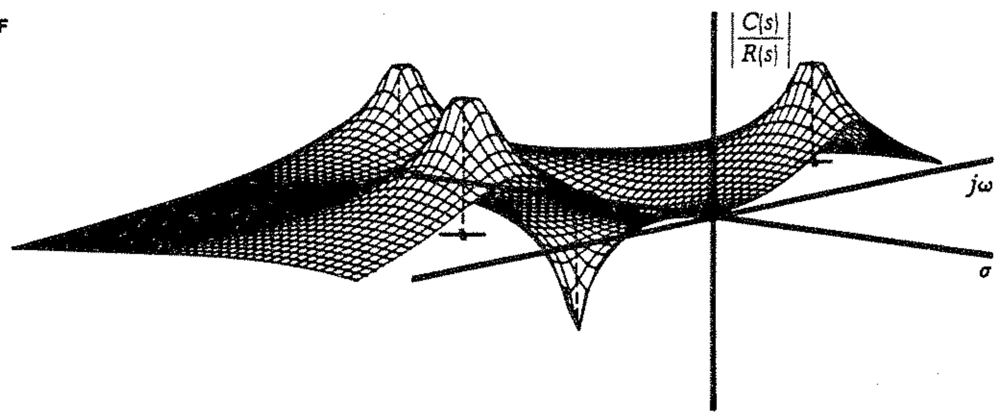
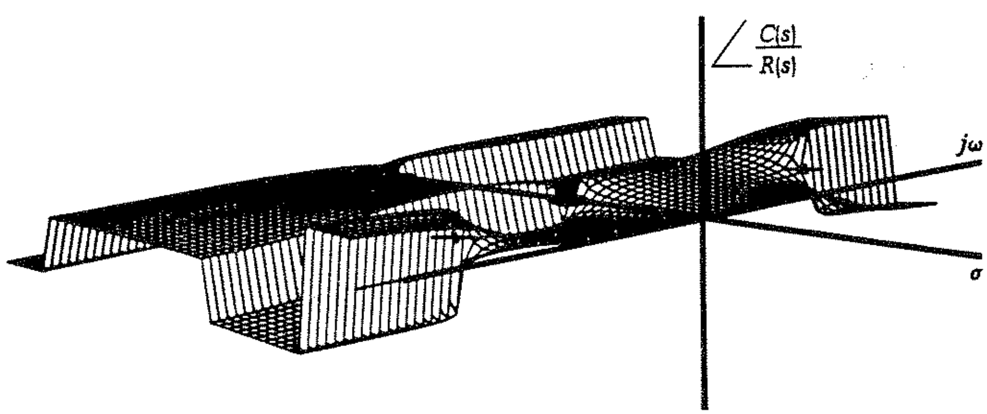


FIGURE 1-8.

PHASE PLOT OF $\dfrac{G(s)}{1 + GH(s)}$ **VERSUS VALUES OF** s



3-2.2: Nyquist's Stability Criterion (*s*-plane)

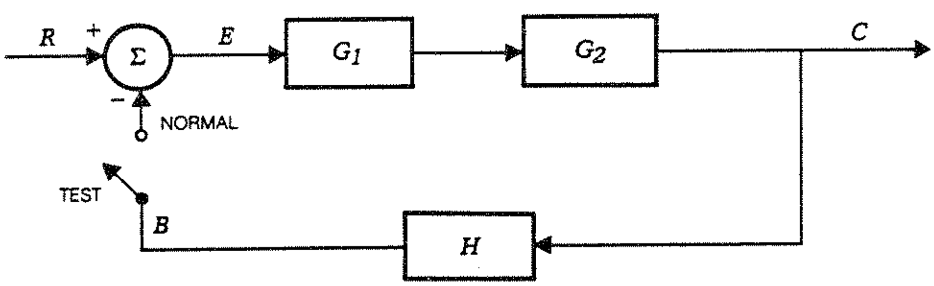
With the evaluation of transfer functions over the *s*-plane well established, the fundamental condition for stability discovered by Nyquist, can now be presented. Simply stated, for a control system to be stable, there can be no closed-loop poles in the right half of the *s*-plane. (Poles on the $j\omega$ axis are not directly addressed but are generally considered to represent instability.) This relationship between closed-loop pole locations and system stability constitutes Nyquist's Stability Criterion as applied to the *s*-plane. This relationship can be extremely useful in predicting the stability of a system if the position of each closed-loop pole is known. Trying to determine the exact location of closed-loop poles from measured data without a computer, however, can often be a difficult task. Fortunately, Nyquist's original work included a very useful technique for evaluating the presence of closed-loop poles in the right-half plane without necessarily knowing their exact locations. To examine this technique closely, however, we will need a few more terms and definitions.

In the preceding paragraphs it was established that the roots of the equation $1 + GH(s) = 0$ (i.e. values of *s* for which $GH(s) = -1$) were the closed-loop poles and the sole factor in determining if the system would be stable. Because of its influence on the stability of the system and, ultimately, the character of the time domain response, the equation $1 + GH(s) = 0$ is known as the *characteristic equation*.

From the characteristic equation it is apparent that the term $GH(s)$ contains all the information concerning the location of the closed-loop poles ($GH(s)$ is understood to represent the transfer function of all of the elements in the loop between the error signal (E) and the feedback signal (B)). The function $GH(s)$ is called the *loop transfer function* or *open-loop transfer function* and is denoted by either $GH(s)$ or $B(s)/E(s)$, as shown in Figure 1-9. This note uses the notation $GH(s)$ and refers to it as the open-loop transfer function.

FIGURE 1-9.

OPEN-LOOP TRANSFER FUNCTION OF A CLOSED-LOOP CONTROL SYSTEM



$GH(s) = G_1G_2H(s) = \frac{B(s)}{E(s)}$ FOR SWITCH IN "TEST" OR OPEN POSITION (THEREFORE THE NAME "OPEN-LOOP TRANSFER FUNCTION")

ALSO: $B(s) = E\,G_1\,G_2\,H(s)$ FOR SWITCH IN EITHER "TEST" OR "NORMAL" POSITION

$$\frac{B(s)}{E(s)} = G_1\,G_2\,H(s)$$

At this point it is worthwhile to recognize that $G(s)$ and $H(s)$ are themselves generally ratios of polynomials in s. $G(s)$ and $H(s)$ can therefore be represented by:

$$G(s) = \frac{G_n(s)}{G_d(s)} \text{ and } H(s) = \frac{H_n(s)}{H_d(s)}$$

where the subscripts n and d indicate the numerator and denominator portions of $G(s)$ and $H(s)$, respectively. If the closed-loop transfer function is reformulated in terms of the numerator and denominator of $G(s)$ and $H(s)$ we have:

$$\frac{C(s)}{R(s)} = \frac{G(s)}{1 + GH(s)} = \frac{\dfrac{G_n(s)}{G_d(s)}}{1 + \dfrac{G_n(s)H_n(s)}{G_d(s)H_d(s)}} = \frac{\dfrac{G_n(s)H_d(s)}{G_d(s)H_d(s)}}{\dfrac{G_d(s)H_d(s) + G_n(s)H_n(s)}{G_d(s)H_d(s)}}$$

The objective of expressing the closed-loop transfer function in this manner is to illustrate that the term $1 + GH(s)$ itself has poles and zeros, and that it is the zeros of this term that determine the poles of the closed-loop transfer function. It is also worth noticing that the zeros of the closed-loop transfer function are the roots of the equation $G_n(s)H_d(s) = 0$.

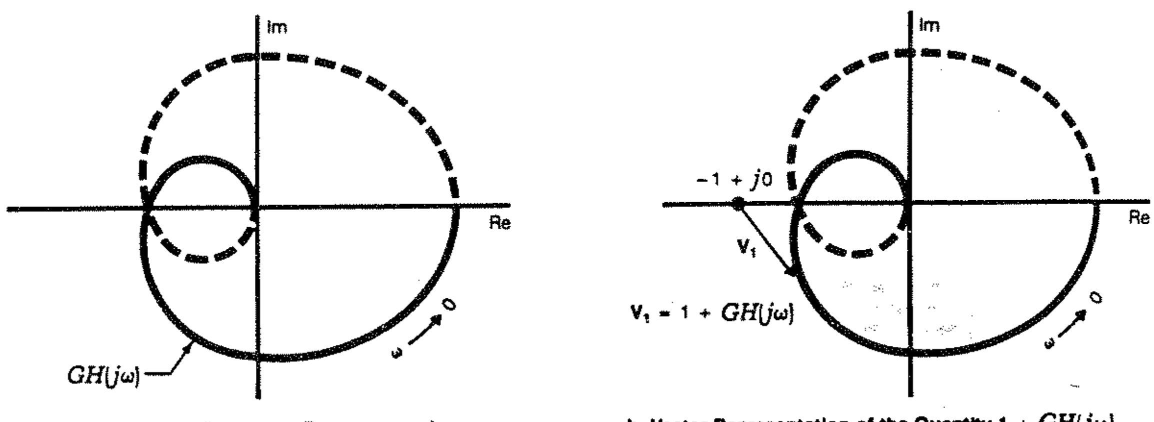
3-2.3: Nyquist Diagrams

It was Nyquist's observation that the frequency response of the open-loop transfer function (i.e. $GH(j\omega)$) can be used to determine if there are any zeros of the term $1 + GH(s)$ (and therefore poles of the closed-loop transfer function) in the right half of the s-plane. To make this determination, $GH(j\omega)$ is first plotted on a two-dimensional Cartesian coordinate system whose ordinate is the imaginary part of $GH(j\omega)$ and abscissa is the real part of $GH(j\omega)$. The complex conjugate of the frequency response curve is then plotted on the same graph, as shown by the dashed line in Figure 1-10a.

The next step is to establish a vector $\mathbf{V_1}$ whose tail is affixed to the point $-1 + j0$. If the head of the vector is then placed anywhere along the curve of $GH(j\omega)$, the vector then represents the quantity $1 + GH(j\omega)$, as illustrated in Figure 1-10b.

FIGURE 1-10.

NYQUIST DIAGRAM IN RECTANGULAR COORDINATES

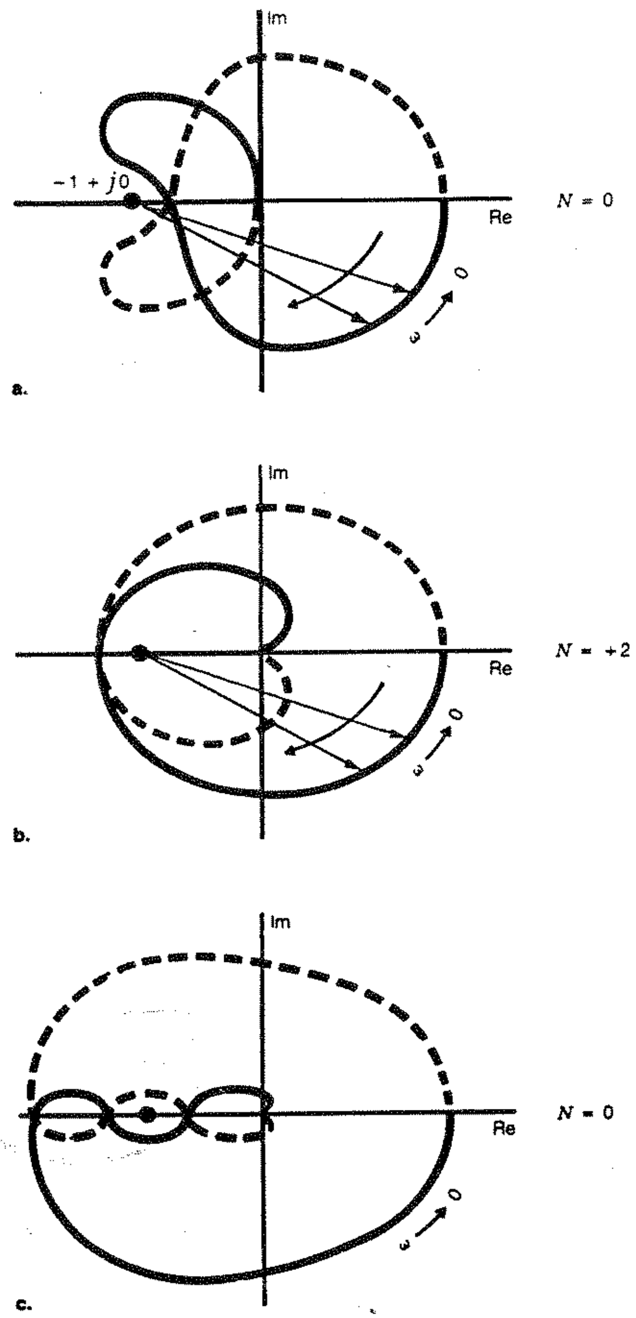


a. Plot of Open-Loop Frequency Response and Its Complex Conjugate.

b. Vector Representation of the Quantity 1 + $GH(j\omega)$.

FIGURE 1-11.

NYQUIST DIAGRAMS SHOWING
a. STABILITY,
b. INSTABILITY AND
c. CONDITIONAL STABILITY



AS ILLUSTRATED, THE SYSTEM IS STABLE.

HOWEVER, IF THE GAIN IS INCREASED SO THAT THE AREA SHADED IN BLUE ENCLOSES THE −1 + $j0$ POINT, THEN $N = 2$, AND THE SYSTEM IS UNSTABLE.

ALSO, IF THE GAIN IS DECREASED SO THAT THE AREA SHADED IN GRAY ENCLOSES THE −1 + $j0$ POINT, THEN $N = 2$, AND THE SYSTEM IS AGAIN UNSTABLE.

THEREFORE BY EITHER INCREASING OR DECREASING THE GAIN, THE SYSTEM BECOMES UNSTABLE (I.E.,THE SYSTEM IS CONDITIONALLY STABLE).

3-2.4: Nyquist's Stability Criterion (Nyquist diagram)

At this point, Nyquist's Stability Criterion states that as the head of the vector traces the $GH(j\omega)$ curve in the direction of increasing positive frequency, the net number of complete rotations N is equal to the number of poles P_r of the term $1 + GH(s)$ in the right half of the s-plane minus the number of zeros Z_r of the term $1 + GH(s)$ in the right half of the s-plane. That is:

$$N = Z_r - P_r$$

where N is positive for clockwise rotations and negative for counterclockwise rotations. We therefore know that a system is stable only if $N = -P_r$. It is a general consensus that for most real systems $P_r = 0$ and, therefore, $N = Z_r$. When this assumption is true, the condition for stability can be restated as: a system is stable if and only if $N = 0$. Figure 1-11 illustrates examples of systems which are stable, conditionally stable, and unstable.

3-2.5: Magnitude and Phase Contours

The Nyquist diagram can also be used to evaluate the closed-loop frequency response from the open-loop frequency response if the system being analyzed has unity feedback. For $H(j\omega) = 1$ the closed-loop transfer function for real frequencies becomes:

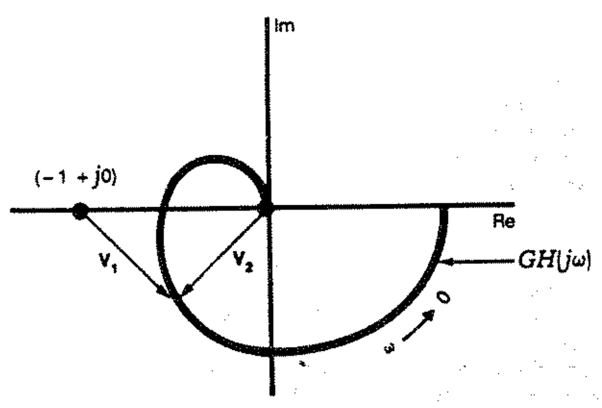
$$\frac{C(j\omega)}{R(j\omega)} = \frac{G(j\omega)}{1 + G(j\omega)}$$

If another vector $\mathbf{V_2}$ is added to the Nyquist diagram so that it projects from the origin and meets with the vector $\mathbf{V_1}$ at the curve of $G(j\omega)$, then the closed-loop transfer function can be represented by the ratio of $\mathbf{V_2}/\mathbf{V_1}$, as shown in Figure 1-12.

Useful tools for evaluating the performance of a unity-feedback control system are *magnitude contours* (often referred to as *M-contours*). A magnitude contour is a locus of points for which the ratio of the magnitudes of $\mathbf{V_1}$ and $\mathbf{V_2}$ is a constant. When plotted on the Nyquist diagram, a magnitude contour will appear as a circle (except when $|\mathbf{V_2}|/|\mathbf{V_1}| = 1.0$), as shown in Figure 1-13. When the open-loop transfer function is plotted on a Nyquist diagram with magnitude contours, the maximum gain of the closed-loop transfer function can be identified as the value of the magnitude contour which is tangent to the plotted curve, as shown in Figure 1-14. A similar diagram can also be constructed for constant values of phase difference between $\mathbf{V_1}$ and $\mathbf{V_2}$. Plots of constant phase are called *phase contours* or *N-contours*.

FIGURE 1-12.

EVALUATION OF CLOSED-LOOP FREQUENCY RESPONSE FROM OPEN-LOOP FREQUENCY RESPONSE (UNITY FEEDBACK)



$\mathbf{V_1} = 1 + GH(j\omega)$

$\mathbf{V_2} = GH(j\omega)$

FOR $H(j\omega) = 1$

$\mathbf{V_1} = 1 + G(j\omega)$

$\mathbf{V_2} = G(j\omega)$

THEREFORE: $\frac{\mathbf{V_2}}{\mathbf{V_1}} = \frac{G(j\omega)}{1 + G(j\omega)}$ = CLOSED-LOOP FREQUENCY RESPONSE FOR A UNITY FEEDBACK CONTROL SYSTEM

3-2.7: Gain Margin and Phase Margin

Bode also explained that an open-loop frequency response curve which just met this criterion would rarely produce a stable system since any small variations in the system's performance would place the response in an unstable region. He therefore suggested that a certain amount of margin should be allotted for both the phase and gain values as they approached the point representing a magnitude of 1 and a phase shift of −180 degrees. These margins are now standard performance parameters known as the phase margin and gain margin.

Phase margin is defined as 180 degrees minus the absolute value of the phase of the open-loop frequency response at the point where the magnitude of the open-loop frequency response (i.e., the open-loop gain) is equal to one. That is:

$$\text{phase margin} = 180 - |\angle GH(j\omega)| \text{ where } |GH(j\omega)| = 1$$

Gain margin is defined as the reciprocal of the open-loop frequency response gain at the point where the phase of the open-loop frequency response is equal to minus 180 degrees. That is:

$$\text{gain margin} = \frac{1}{|GH(j\omega)|} \quad \text{where } \angle GH(j\omega) = -180 \text{ degrees}$$

The gain margin therefore represents the amount the open-loop gain can be increased before it reaches a magnitude of 1. Examples of gain margin and phase margin are shown in Figure 1-16.

The importance of the gain and phase margin can be fully appreciated when they are compared with, and shown to correlate with, the time domain parameters of the step response. For example, for a system whose response characteristics are dominated by a pair of complex poles (a very common case), the following relationships can be observed. An increase or decrease in the system's frequency independent gain[1] will cause both the gain margin and phase margin to decrease or increase, respectively. For the case in which the gain is increased, the following events will occur:

- the gain margin and phase margin will decrease
- the maximum overshoot will increase
- the rise time will decrease
- and, in some cases, the steady-state deviation will decrease

From this series of interactions it can be seen that the development of a control system is generally a trade-off between the desired performance characteristics. Although each control system has unique requirements, minimum acceptable levels of gain margin and phase margin are typically 2 (or 6 dB)[2] and 30 degrees, respectively.

[1] Frequency independent gain is also referred to as proportional amplification and is represented by the variable K. A more detailed explanation is provided in the discussion of the root locus diagram, Section 4-4.

[2] dB represents a unit of comparison known as the decibel. It is calculated for both voltage and power ratios with respective formulas for each being: dB = 10 log (power ratio) and dB = 20 log (voltage ratio). See Hewlett-Packard Application Note 243, *The Fundamentals of Signal Analysis*, p. 5, for further details.

FIGURE 1-16.

MEASURING GAIN MARGIN AND PHASE MARGIN ON A NYQUIST DIAGRAM

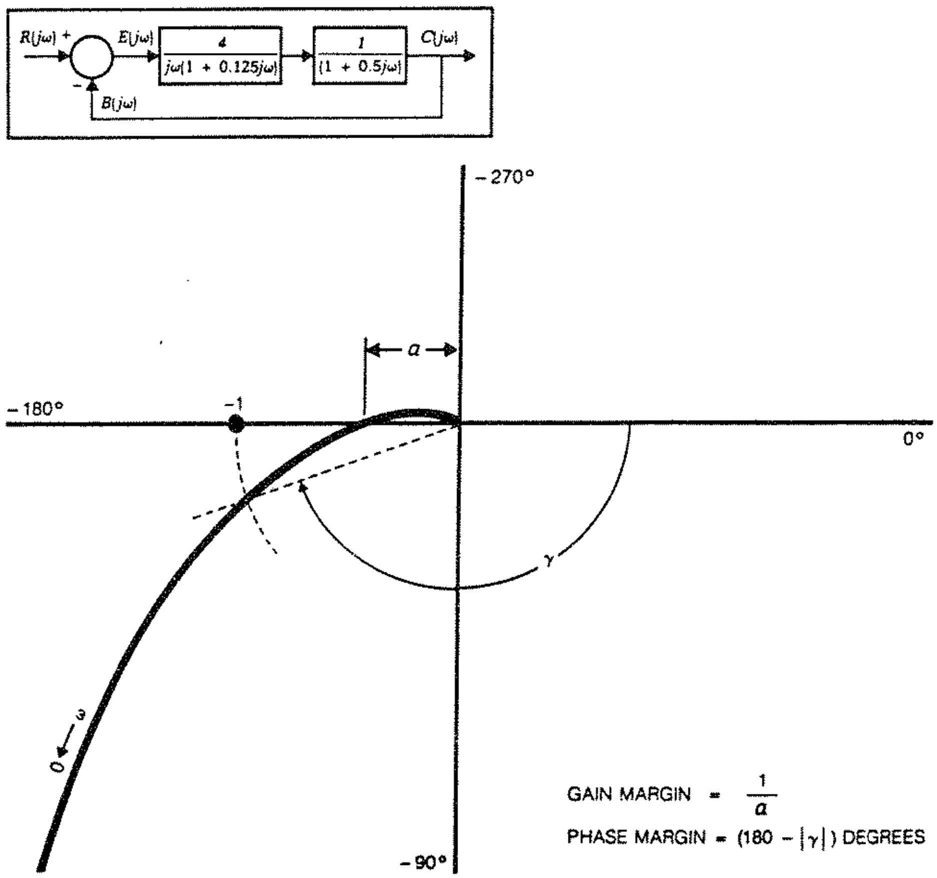
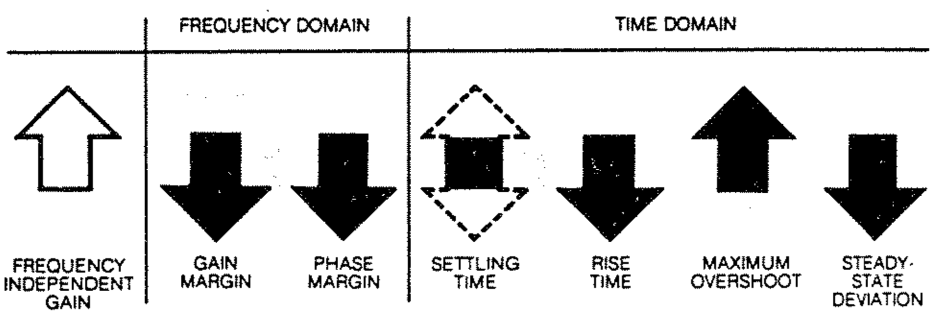


TABLE 1-1.

GENERALIZED RELATIONSHIPS BETWEEN TIME DOMAIN AND FREQUENCY DOMAIN PERFORMANCE PARAMETERS RELATIVE TO AN INCREASE IN THE FREQUENCY INDEPENDENT GAIN



[1] Figure 1-16 is adapted from the American National Standard ANSI MC85. 1M-1981, Terminology for Automatic Control.

In addition to gain margin and phase margin, there are several other performance quantities such as the system type and steady-state error coefficients which can be extracted from a Nyquist diagram. Unfortunately, a complete description of these quantities is beyond the scope of this document (several references for further study are listed at the end of this note). It can be assumed, however, that the key performance characteristics of a control system can be adequately characterized with a Nyquist diagram.

One shortfall of the Nyquist diagram is the difficulty encountered when attempting to predict the effects of changes to a control system. Most alterations (other than a change in frequency independent gain) require a significant number of calculations, or a new measurement, to accurately obtain the correct Nyquist diagram. As a result, several other analysis techniques were developed to make the design and analysis of a control system easier. These are discussed in detail in the following chapter.

Chapter 4: More Tools for Design and Analysis

There is perhaps no design tool which has gained as much popularity as the diagram which Bode presented in his 1940 paper, "Relations Between Attenuation and Phase in Feedback Amplifiers." This chapter looks at the famous *Bode diagram* and two other popular design and analysis tools: the Nichols diagram and root locus diagram.

4-1: The Bode Diagram

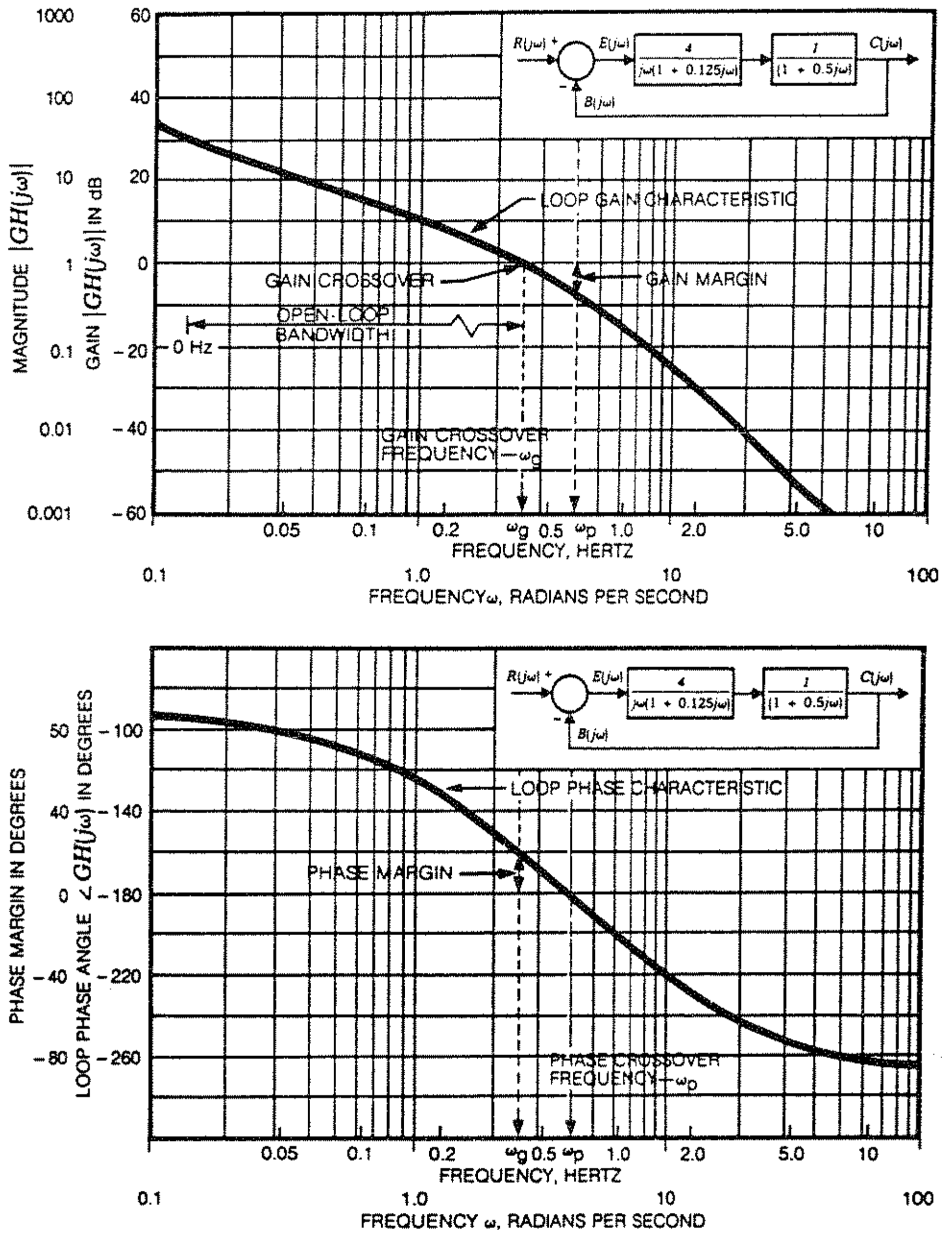
The Bode diagram is similar to the Nyquist diagram in that it also represents a plot of the open-loop frequency response. However, the Bode diagram considers the gain and phase of the response separately by providing a plot of each versus frequency. The plot of open-loop gain versus frequency is called the *loop gain characteristic* and the plot of open-loop phase versus frequency is called the *loop phase characteristic*, as shown in Figure 1-17.

Bode diagrams use logarithmic units (i.e., dB) for gain and logarithmic scales for frequency; phase is the only parameter represented linearly. The use of logarithmic scales and units provides the Bode diagram with three key advantages. First, by displaying gain in units of dB, a much wider range of gain levels can be displayed on a single plot. Second, the effect on the open-loop frequency response of adding a new component in a control loop can be calculated through simple addition rather than multiplication. That is, by plotting the frequency response of a new component on the same Bode diagram as the original response, the frequency response of the new system can be calculated by graphically adding the two plots. Third, the logarithmic scales and units facilitate a technique for quickly estimating the frequency response of an analytic transfer function. This last point is a major topic of Bode's paper. In his paper, Bode presented a relatively simple set of procedures for constructing a set of curves which would closely estimate the actual frequency response of a transfer function without ever actually calculating or measuring the response.

An equally powerful tool was the ability to apply Bode's construction procedures in reverse. That is, to obtain information about the analytic transfer function from the measured frequency response.

FIGURE 1-17.

BODE DIAGRAMS SHOWING FREQUENCY RESPONSE FOR A TYPICAL OPEN-LOOP TRANSFER FUNCTION



[1] Figure 1-17 is adapted from the American National Standard ANSI MC85. 1M-1981, Terminology for Automatic Control.

4-2: Stability and the Bode Diagram

The Bode diagram also provides a simple check for stability. According to Bode's interpretation of Nyquist's Stability Criterion, for a system to be absolutely stable, the loop gain characteristic must be less than one before the loop phase characteristic exceeds (becomes more negative than) 180 degrees. On a Bode diagram, this means the frequency at which the loop gain characteristic becomes equal to 0 dB (i.e., the *gain crossover frequency*) must be lower than the frequency at which the loop phase characteristic becomes equal to −180 degrees (i.e., the *phase crossover frequency*).

The phase margin, gain margin, and open-loop bandwidth[1] of a system can also be read directly from the Bode diagram, as shown in Figure 1-17.

One of two disadvantages of the Bode diagram is that there is no technique for directly relating the open-loop frequency response to the closed-loop frequency response (as was possible with the magnitude and phase contours of the Nyquist diagram). However, the frequency response information from a Bode diagram can be directly transferred to a Nyquist or Nichols diagram to evaluate the closed-loop frequency response. (It is important to note that a reverse exchange of information, that is, from a Nichols or Nyquist diagram to a Bode diagram, may not be possible due to the loss of frequency information in both the Nichols and Nyquist diagrams.)

A second disadvantage of the Bode diagram is its limited ability to verify the stability of control systems which are conditionally stable. Fortunately, conditionally stable systems are rarely designed intentionally and can be analyzed by transferring the frequency response data to a Nyquist diagram if necessary.

4-3: The Nichols Diagram

The Nichols diagram (also known as the *log magnitude-angle diagram*) is essentially a combination of the Nyquist and Bode diagrams. It is conceptually similar to the Nyquist diagram in that it plots the magnitude of $GH(j\omega)$ versus the angle of $GH(j\omega)$ as a function of frequency (ω) on a single graph, as shown in Figure 1-18. Its structure, however, more closely resembles a Bode diagram in that it uses a rectangular coordinate system and scales gain in units of dB.

The Nichols diagram incorporates some of the advantages provided by the Bode and Nyquist diagrams into a single graph. By plotting gain versus phase, the Nichols diagram allows the construction of magnitude and phase contours similar to those used on the Nyquist diagram. However, by scaling the gain in units of dB, a single set of contours can be applied over a much broader range of gain levels. A single Nichols diagram can therefore provide a direct readout of the closed-loop frequency response (of a unity feedback control system) for a much broader range of open-loop gains. Nichols diagrams which have a large set of magnitude and phase contours drawn on them are often called *Nichols charts*.

[1] Open-loop bandwidth is defined as the frequency span between 0 Hz and frequency at which the gain of the open-loop frequency response is equal to 1.

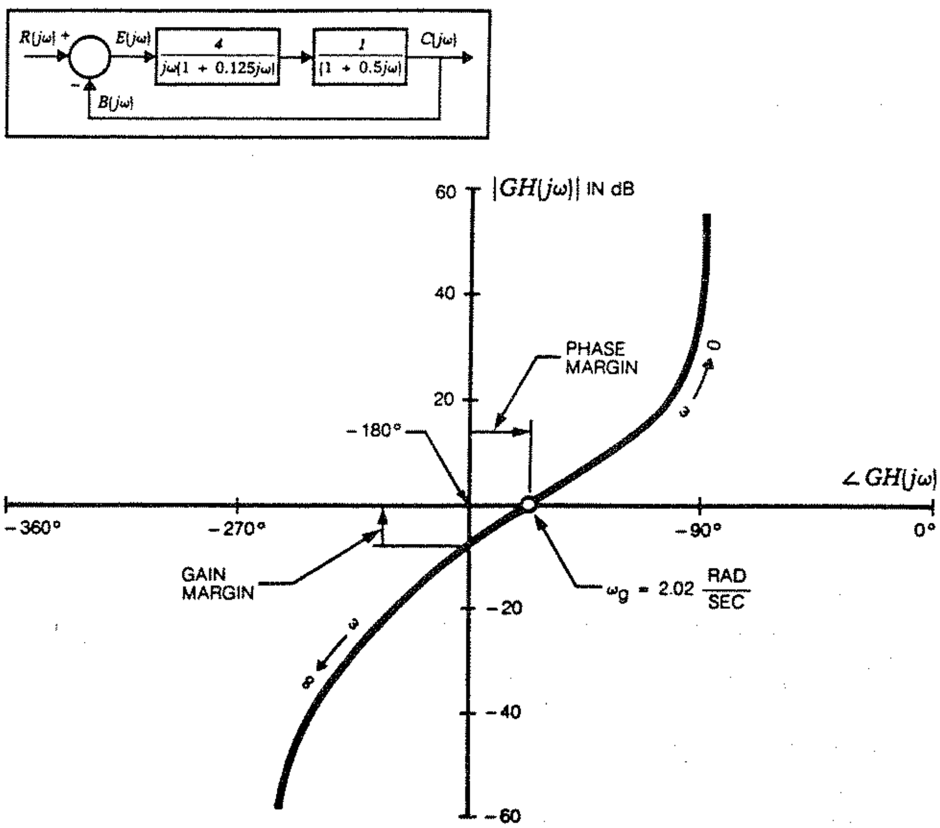
Gain margin and phase margin can also be read directly from the Nichols diagram. However, to obtain the open-loop bandwidth, the gain crossover frequency must be evaluated while the plot is being constructed, and then marked on the graph, as shown in Figure 1-18.

The main disadvantage of the Nichols diagram is the difficulty in plotting $GH(j\omega)$ directly from the transfer function. Unlike the Bode diagram, there is no simple set of rules which provides a quick estimation of a transfer function's frequency response. It is therefore difficult to predict the effect of a compensation circuit on the system's performance.

The Nichols diagram is also limited in its ability to verify the stability of conditionally stable systems. However, like the Bode diagram, the frequency response information can be transferred to the Nyquist diagram for analysis.

FIGURE 1-18.

NICHOLS DIAGRAM (LOG MAGNITUDE VERSUS ANGLE DIAGRAM) FOR A TYPICAL OPEN-LOOP TRANSFER FUNCTION



4-4: The Root Locus Diagram

The root locus diagram (or root locus plot) was developed by W.R. Evans and presented in his 1950 paper, "Control System Synthesis by Root Locus Method"[1]. The root locus diagram is a departure from the frequency response plotting techniques used by the Bode, Nichols and Nyquist diagrams. All three of the latter techniques use the frequency response of the open-loop transfer function, $GH(j\omega)$, to gain information about the relative location of the closed-loop poles in the s-plane. The root locus diagram, however, uses the location of the open-loop poles and zeros in the s-plane to predict the actual location of the closed-loop poles. Before discussing the root locus diagram further, it is again necessary to introduce another concept.

The symbol G was previously defined as a transfer function whose gain and phase characteristics change with respect to the variable s or $j\omega$. It can, however, be divided into two factors: 1) a proportional amplification often denoted as K, which is independent of s or $j\omega$ and associated with a dimensioned scale factor relating the units of input and output; 2) a dimensionless factor often denoted as G which is dependent on s or $j\omega$. Therefore, if K is used as a prefix when expressing a transfer function, it is understood that K represents a gain value extracted from the transfer function which is independent of s or $j\omega$. For example, if the open-loop transfer function is expressed as $KGH(j\omega)$, it is understood that K is the gain portion of $GH(j\omega)$ which is independent of $j\omega$.

The objective of the root locus diagram is to graphically locate values of s which set the open-loop frequency response equal to -1, that is s such that $GH(s) = -1$. These values of s will therefore also represent roots of the characteristic equation $1 + GH(s) = 0$ and, further, represent the location of the closed-loop poles.

The power of the root locus technique is its recognition of the frequency independent gain of the open-loop transfer function, K of $KGH(s)$. The root locus technique recognizes that for each value of K there is a unique set of values for s which satisfy the equation $KGH(s) = -1$. For example, if K is set equal to 3 in the open-loop transfer function:

$$KGH(s) = \frac{K}{s(1+0.125s)(1+0.5s)}$$

then there exists a unique set of values of s, in this case those shown in Figure 1-19a, for which $3GH(s) = -1$ (or alternatively, $GH(s) = -1/3$). If K is set equal to 4, then there exists another set of values of s for which $GH(s) = -1/4$, as illustrated in Figure 1-19b. This new set of values for s represents the new locations of the closed-loop poles when K is increased from 3 to 4.

If the unique set of values for s were calculated for each value of K from zero to infinity and plotted on the same graph, the result would be a set of lines which represent a locus of roots to the equation $1 + KGH(s) = 0$ for all possible values of K, as shown in Figure 1-19c. This plot is called a *root locus diagram*.

If root locus diagrams were constructed in this fashion, it would require many calculations and make the construction of the diagram much too involved to be of practical value, at least without the aid of a computer. Fortunately, Evans also presented a technique for graphically estimating the root locus diagram based on the location of the open-loop poles and zeros in the s-plane. The procedure is relatively simple and it is not uncommon for people who have mastered the root locus technique to quickly sketch the root locus diagram based solely on the location of the open-loop poles and zeros (i.e., with virtually no calculations). A root locus diagram will therefore generally include designators indicating the position of the open-loop poles and zeros as shown in Figure 1-20.

The root locus diagram is a very powerful design tool since it works directly with the location of the closed-loop poles in the s-plane. However, the root locus technique can only be used if the number and location of the open-loop poles and zeros are known. It is therefore less flexible than the Nyquist or Bode diagrams which need only the measured open-loop frequency response to predict performance and provide design information. It does, however, provide more information during the initial design process and is better suited for the design of complex compensation networks.

1 "Control System Synthesis by Root Locus Method," *Trans. AIEE*, 69, 1-4 (Mar 10, 1950).

FIGURE 1-19.

VALUES OF s WHICH SATISFY THE EQUATION

$$GH(s) = \frac{K}{s(1 + 0.125s)(1 + 0.5s)} = -1$$

FOR a. $K = 3$,
b. $K = 4$,
c. $0 \leq K \leq \infty$

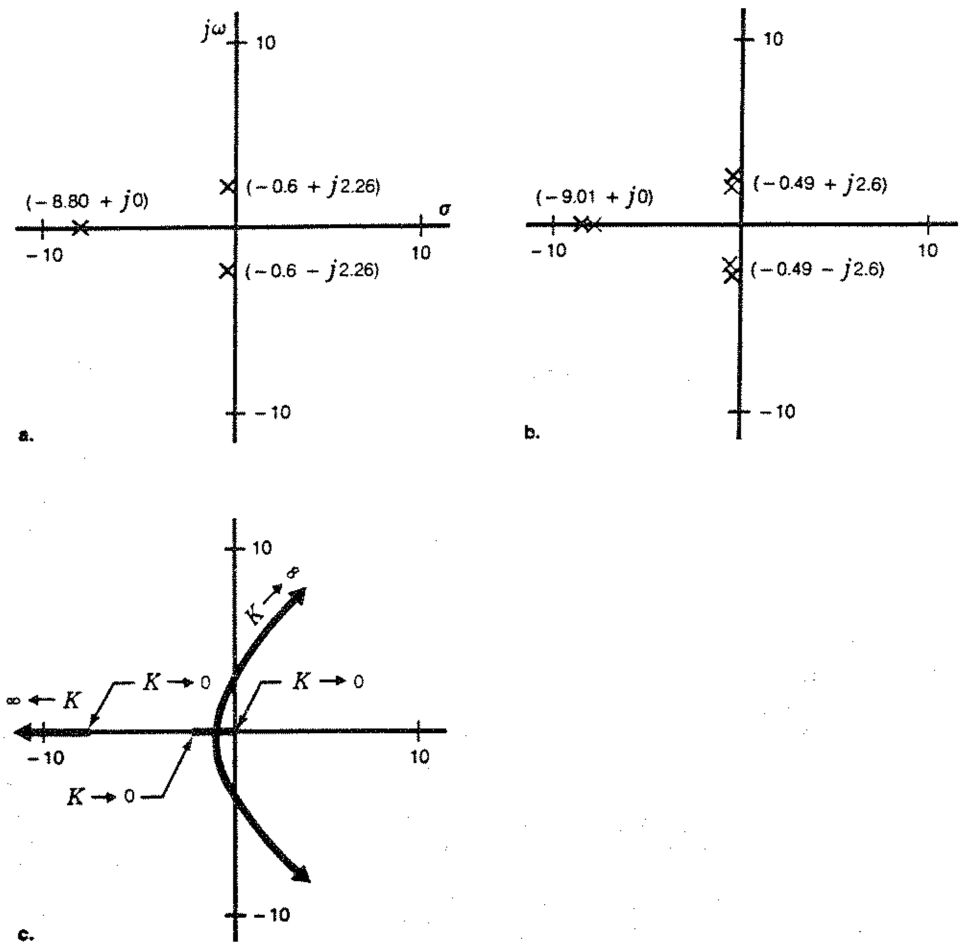
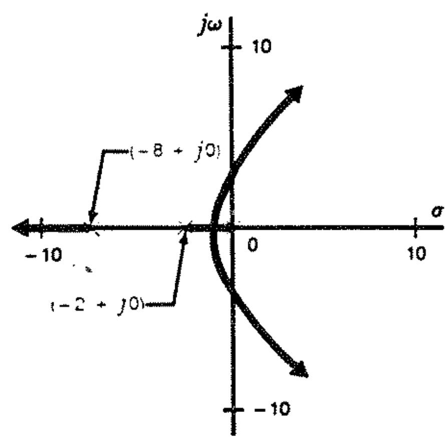


FIGURE 1-20.

ROOT LOCUS DIAGRAM OF THE EQUATION

$$GH(s) = \frac{K}{s(1 + 0.125s)(1 + 0.5s)} = -1$$



Chapter 5: Nonlinear Systems

The design and analysis tools presented so far have all assumed that the control system or subsystem being analyzed is linear. Unfortunately, the vast majority of control systems are actually nonlinear, either by design or by virtue of the components within the system.

There are some very complex analysis tools which deal directly with nonlinearities; however, a very common practice is to obtain an approximation of the system's nonlinear operation which best conforms to a linear response. The approximation can then be used with the tools presented in the previous chapters.

For example, Figure 1-21a shows a typical gain curve (V_{OUT}/V_{IN}) which is essentially linear for input voltages less than V_L and nonlinear for input voltages greater than V_L.

If the system characterized by Figure 1-21a is operated within a narrow range of voltages centered about a voltage V_1 as shown in Figure 1-21b, then the system will operate over a linear region of the curve and can be modeled with the linear equation:

$$V_{OUT} = aV_{IN}$$

or $$\frac{V_{OUT}}{V_{IN}} = a$$

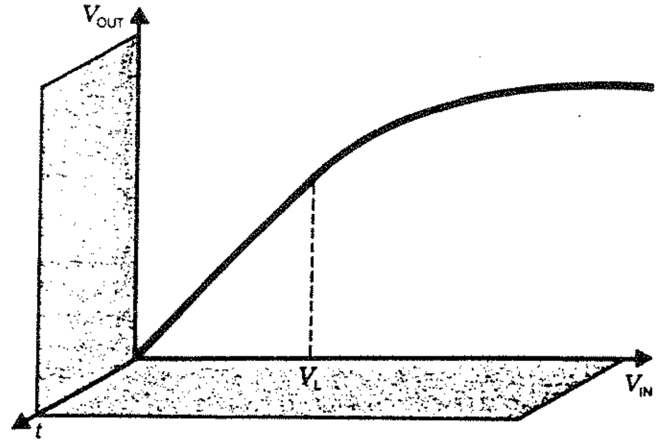
where a is a constant.

If, however, the system operates under the same conditions except at a higher average voltage V_2 as shown in Figure 1-21c, then the system is not operating in a linear region and a linear approximation is required.

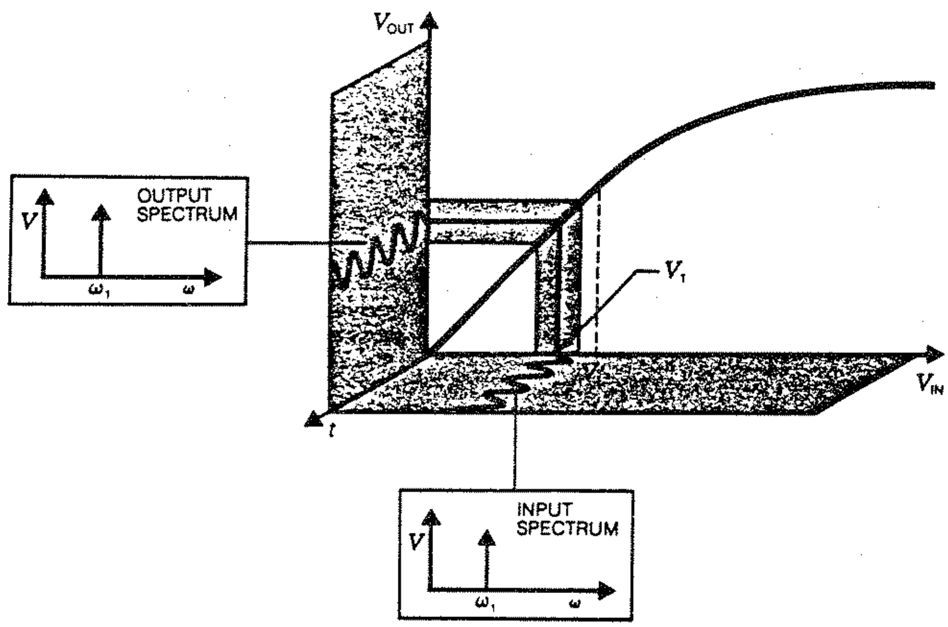
Graphically, a linear approximation could be obtained by simply drawing a straight line through the operating region which best fits the gain curve. This approximation, however, would not address the distribution of energy throughout the response spectrum due to the distortion of the output waveform, as shown in Figure 1-21c.

For this type of nonlinearity, a better technique for obtaining a linear approximation of the system's gain is to measure only that part of the response spectrum which is at the same frequency as the input. That is, measure the system gain at the fundamental frequency of the stimulus and ignore all the other frequency components, including those created by system nonlinearities. If a series of both gain and phase measurements are made over a range of frequencies, the results can be plotted to produce a graph of the system's frequency response. The resulting frequency response can then be used to generate a transfer function based solely on the fundamental. Such a transfer function is often called a *describing function* and is generally considered a good linearized approximation of a system with nonlinearities such as harmonic distortion and intermodulation distortion.

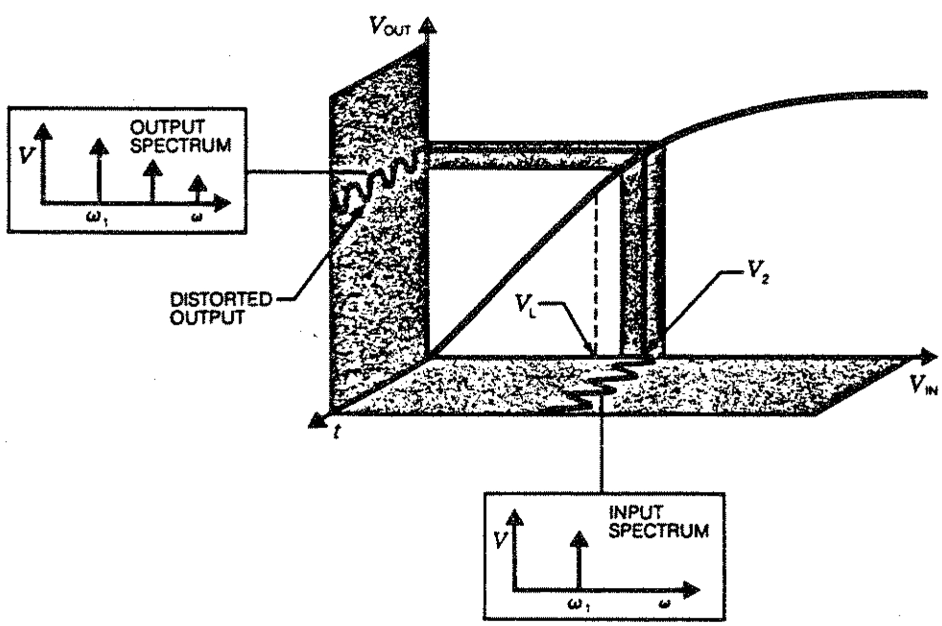
FIGURE 1-21.



a. Typical Gain Curve with Additional Third Axis Representing Time.



b. Operation within Linear Region.



c. Operation in Non-Linear Region.

A common technique used to make the measurement described is to stimulate the system with a swept sine wave source and measure both the stimulus and the response with narrow bandpass filters which track the frequency of the source. Test instruments capable of making this type of measurement include network analyzers, frequency response analyzers, and properly equipped Dynamic Signal Analyzers (DSAs).

It is important to note that for any small change in either the mean voltage V_2 or the amplitude of the sine wave itself, the measured frequency response will also change. This change in measurement result due to changes in the testing conditions is a common phenomenon associated with most nonlinear devices.

If a nonlinear system is both sensitive to changes in the stimulus signal (as described above) and operated over a wide range of stimulus levels, then there is typically no one unique frequency response or describing function which can accurately model the operation of the system.

As a practical solution to this problem, a nonlinear device is typically tested under conditions which closely approximate the actual operating conditions of the system. If the operating conditions themselves do not vary widely, and they can be adequately simulated during testing, then the resulting measurements are generally assumed to be a linearized estimation of the device's operation.

To provide maximum flexibility in obtaining a linearized estimation of a device's operation, advanced DSAs provide two separate analysis functions for measuring the frequency response of both linear and nonlinear devices: Swept Fourier Analysis (SFA) and Fast Fourier Transform (FFT) analysis. More information concerning SFA and FFT analysis as well as many of the other measurement capabilities provided by DSAs are presented in Part 2 of this application note.

Part 2

Measurement and Analysis Tools Applied to the Development Process

Historically, a test instrument's primary contribution to the development of a control system has been the collection of stimulus and response data. While this is still true, microprocessor-based Dynamic Signal Analyzers (DSAs) have expanded the role of the test instrument to include significant contributions in other areas of control system development, such as modeling and design.

The purpose of the following chapters is to provide a basic introduction to the measurement and analysis capabilities provided by high performance DSAs, and to suggest how these tools can be used in the various phases of control system development.

Chapter 1: Modeling the Development Process

In general, it is recognized that the development of a control system typically involves some unique combination of five distinct processes: model, design, build, test and analyze. For the purpose of this application note, these five processes are defined as follows:

Design — *determining the combination of physical or theoretical components or parameters that will produce a desired action or result.*

Model — *the process of transforming the observed characteristics of some device or process into theoretical representations consistent with the analysis/design technique being used.*

Build — *the physical construction of a system and/or its components.*

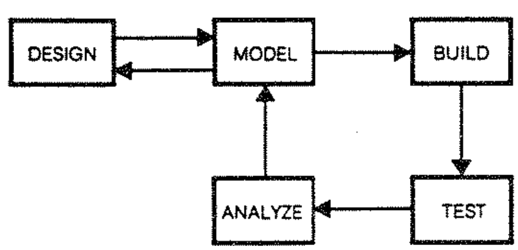
Test — *the collection of stimulus and/or response data.*

Analyze — *determining the value of parameters, either physical or theoretical, used to characterize the action or function of a device. Also establishing the relationships, if any, between those parameters.*

When grouped into a process flowchart, these five processes can be used to model the development of a control system. A generalized example of a "development process" model/flowchart is shown in Figure 2-1.

To emphasize the DSA's ability to contribute throughout the development of a control system, the following chapters examine the tasks associated with each development process (with the exception of build) and present the tools provided by DSAs for accomplishing those tasks. To provide a structured introduction, the chapters are presented in the following order: Test, Analyze, Model and Design.

FIGURE 2-1.



Chapter 2: Test

Test: the collection of stimulus and/or response data.

There are many tests which conform with the above definition; however, the most common control system tests are the measurement of a system's response to a step change in the input (i.e., the step response) and the frequency response of the system and/or any of its components.

Instruments which have commonly been used to perform these tests include frequency response analyzers, network analyzers, waveform recorders, strip-chart recorders, and storage oscilloscopes. Typical control system tests often required at least two of these instruments: one instrument to record time domain data (e.g., the impulse response or step response) and another to record frequency domain data (e.g., the open-loop or closed-loop frequency response).

The high performance DSA, however, is a single instrument capable of providing all the measurement capability needed in the dc to 100 kHz frequency range. Technological advances allow DSA to assume 1 to 3 basic configurations: a waveform recorder for direct measurement of time domain data, a frequency response analyzer (i.e. Swept Fourier Analyzer) for providing frequency domain data, or a Fast Fourier Transform (FFT) analyzer which also provides frequency domain information.

In addition to providing three analyzers within one test instrument, the DSA also provides several signal monitoring functions. These functions allow the DSA to automatically optimize measurement conditions during a test, reducing the need for operator interaction.

The remainder of this chapter presents the DSA's basic capabilities for measuring both time domain and frequency domain data.

2-1: Time Domain Measurements

Time domain measurements require the test instrument to record the reaction of a device in response to some controlled change in the system's input. A measurement is generally considered successful if it records the entire response and allows the operator to examine both the long term trend of the response and the details of any short term events.

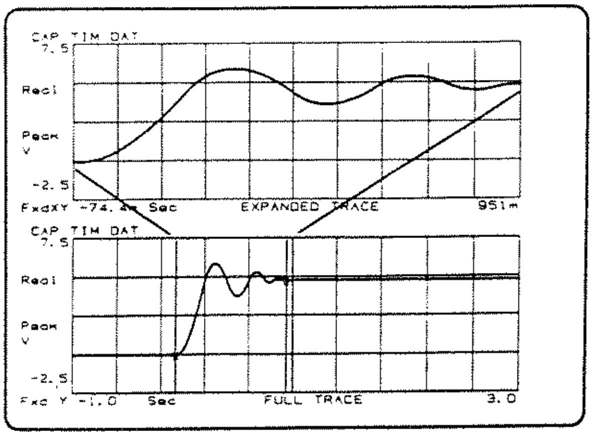
DSAs provide this measurement capability by sampling the signals applied to their inputs and recording the samples as blocks of contiguous data called *time records*. How the time records are stored and how the data within them can be accessed depends on which of two measurement modes, time capture or time throughput, is used to collect the data.

2-1.1: Time Capture

Responses which decay to a steady state value within a few time records can easily be recorded using the DSA's time capture mode. The time capture mode stores a limited number of contiguous time records within the DSA's internal memory. Once collected, all the data can be compressed onto a single trace on the DSA's display. Segments of the compressed data can then be expanded and closely examined on the second trace of the display, as shown in Figure 2-2.

FIGURE 2-2.

A DISPLAY OF A STEP RESPONSE RECORDED WITH TIME CAPTURE SHOWING 10 TIME RECORDS COMPRESSED ON A SINGLE TRACE (LOWER TRACE) AND A PORTION OF THE RESPONSE EXPANDED TO REVEAL DETAIL (UPPER TRACE)



2-1.2: Time Throughput

Occasionally, a device with a very long settling time will require very large amounts of data to be recorded. In these situations, the DSA's time throughput mode can be used to store contiguous time records[1] directly to a mass storage disc without the need for an instrument controller. To study a recorded event, time records are recalled from the disc and presented on the DSA's display.

To ensure that an entire response can be recorded, both time capture and time throughput provide pre- and post-trigger data recording functions. The pre-trigger function allows a specified amount of data obtained before a trigger occurs to be recorded. The post-trigger function allows a specified amount of data to be ignored when obtained after a trigger occurs.

For systems with very fast response times, the pre-trigger function can be used to record the steady-state operation of a system just before a step change is introduced. Alternatively, the post-trigger function can be used to ignore the large amounts of dead time in systems with very slow response times.

In addition to recording and displaying time domain data, DSAs are also capable of recalling recorded data and processing it through a Fast Fourier Transform algorithm. This allows the DSA to provide both time and frequency domain information from one set of recorded data. This capability can be especially valuable for extracting the maximum amount of information from tests which can be performed only once, such as destructive tests.

2-2: Frequency Domain Measurements

Virtually all closed-loop control system development requires the frequency response of the system and/or some of its components to be evaluated by experiment. Unlike most conventional test instruments, advanced DSAs provide two independent techniques for measuring the frequency response of a device: Swept Fourier Analysis and Fast Fourier Transform analysis.

[1] If the DSA collects data much faster than the connected disc can record data, or the DSA collects data faster than it can process the data through its own I/O section, then the time records will not be contiguous. The rate at which time records can be transferred in a contiguous fashion is referred to as the "real-time bandwidth" of the throughput function. More information on real-time bandwidths is available in Hewlett-Packard Application Note 243, *The Fundamentals of Signal Analysis*.

2-2.1: Swept Fourier Analysis

Swept Fourier Analysis (SFA) is a very common measurement technique involving a swept sine wave source and an integration process which emulates a tracking bandpass filter, as shown in Figure 2-3. The primary objective of this measurement technique is to measure the gain and phase shift of a device by measuring only the fundamental component of the stimulus signal and only the fundamental component of the device's response signal (the frequencies of the fundamentals are assumed to be the same). A series of measurements are made at different frequencies to provide a frequency response based on the fundamental of the stimulus and response signals (i.e. ignoring any other spectral components including those generated by nonlinearities such as harmonic distortion).

By using very narrow bandwidths, the effects of nonlinearities such as harmonic distortion, dc offset and random noise can be minimized. This measurement technique also allows those types of nonlinearities which are not affected by narrow filter bandwidths (such as level saturation and frequency shifting of resonances) to be characterized by either making several measurements at different stimulus levels or by sweeping in both directions.

To achieve the narrow filter bandwidths required to measure low frequency systems, DSAs utilize a Discrete Fourier Transform to evaluate the energy within a narrow frequency span. The transform is evaluated at several points during a sweep with the center frequency of the analysis corresponding to the frequency of the swept sine source (thus the term Swept Fourier Analysis). This technique emulates a tracking bandpass filter with very narrow bandwidths, very good harmonic rejection and excellent dc rejection.

An added advantage of using a DSA to make SFA measurements is the availability of automated measurement functions. By constantly monitoring the signals applied to its inputs and referencing past measurements, the DSA can automatically:

- adjust its input sensitivity
- reject measurements in which input overloads occurred
- adjust the frequency resolution of the measurement relative to the rate of change in gain and phase
- repeat a measurement at a given frequency and average the results until an acceptable variance in the measurement is obtained
- adjust the source level to maintain a constant stimulus or response level
- allow the operator to simultaneously monitor the signals applied to the analyzer (in either the time or frequency domains) and view the current measurement.

2-2.2: Fast Fourier Transform Analysis

Compared to SFA, FFT analysis represents more of a parallel approach to measuring a device's frequency response. Rather than sweeping a single bandpass filter as the SFA technique does, the FFT process uses a different form of Fourier integration to create many adjacent bandpass filters (up to 800 in advanced DSAs), as shown in Figure 2-4. These filters selectively and simultaneously measure the energy distributed over an entire frequency span.

A useful analogy is to think of each filter as the bandpass filter of an SFA analyzer. However, rather than collect new data for each measurement point sweep, the FFT process uses time records to collect time domain data and then processes the data through 800 filters simultaneously. This form of parallel processing provides exceptional measurement speeds. It is worth noting, however, that unlike an SFA measurement, an FFT measurement does not filter out energy converted to other frequencies by nonlinearities in the system. Instead, these frequency components (if they are not coherent with the stimulus) are removed by averaging several measurements.

FIGURE 2-3.

SYNCHRONIZED SWEEP OF SINE-WAVE SOURCE AND BANDPASS FILTER CREATED BY FOURIER INTEGRATION

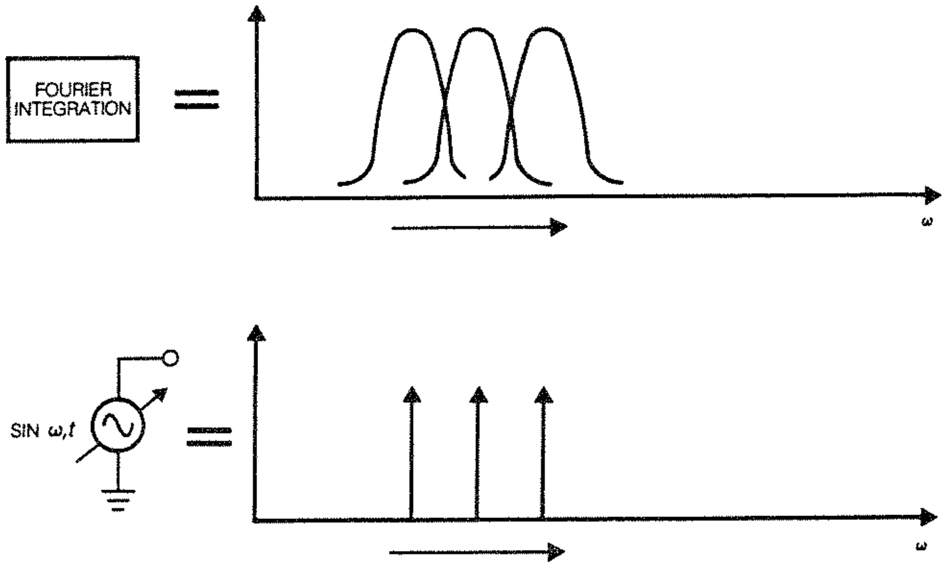
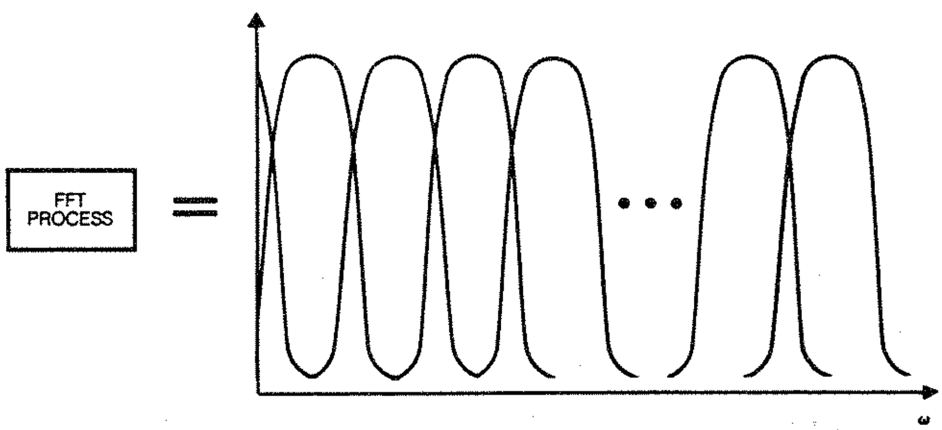


FIGURE 2-4.

MULTIPLE BANDPASS FILTERS PRODUCED BY FAST FOURIER TRANSFORM (FFT) PROCESS



One of the most powerful attributes of the FFT measurement technique is that it allows virtually any type of signal to be used as a stimulus. Common stimulus signals used with FFT measurements include: actual operating signals, sine wave chirps, fixed sine waves, random noise, burst random noise, step functions and impulse functions.

This broad range of stimulus signals increases the resources available for characterizing the operation of a system. Often, selecting the right stimulus signal can provide a better understanding of nonlinearities present in the system and, in some cases, even reduce the overall testing time. The following paragraphs cite some of the benefits offered by certain source types.

An important class of stimulus signals are those stimuli which produce energy at all of the frequencies being analyzed by the FFT algorithm and do so within one time record. Stimuli which meet this criteria (such as the sine chirp, random noise, burst chirp and burst random noise stimuli provided by advanced DSAs) allow the FFT algorithm to provide frequency response information over the entire frequency span being analyzed with just one measurement. If any of these stimuli (with the exception of random noise) are used to test a system which is relatively noise-free and linear, a single time record is often sufficient data to produce an accurate frequency response.

When testing a nonlinear system, selecting a stimulus signal which approximates the signals present during normal operation can provide results which more accurately predict the system's operation. The ability to use a random noise stimulus can be very useful in this respect. For example, random noise superimposed on a dc level often resembles the signals present in a servo system much more than a sine wave superimposed on a dc level.

Signals with random amplitude distribution, such as true random and burst random, can be used to provide an approximation of the frequency response of a system with amplitude nonlinearities. Because random noise is characterized by a random level distribution at a given frequency, a random noise measurement produces a frequency response which represents an average of responses taken at several stimulus levels. When attempting to measure the frequency response of a device with an amplitude nonlinearity such as gain compression, a random noise measurement may provide a better approximation of the device's actual operation than a single swept sine measurement.

A random stimulus signal can also reduce the effects of nonlinearities influenced by the direction of a sine sweep. Such nonlinearities often show up as a change in resonance frequencies corresponding to a change in sweep direction (not to be confused with skewed responses caused by excessive sweep speed). Since random noise continuously produces energy over an entire frequency spectrum, the measurement is not affected by transferring energy from one frequency to another.

Some forms of nonlinearities preclude the use of certain stimulus types. For example, when testing systems with a significant amount of dead zone or hysteresis, such as large gear trains, signals such as random noise can be inappropriate. The waveform of a random signal is typically characterized by many changes in slope and a greater concentration of lower level voltages than high level voltages. This would create a lot of noise in a gear train while producing little output. Instead, a sine wave stimulus which spends more time at higher voltage levels and makes fewer slope transitions may be a much better overall stimulus choice.

The decision of which stimulus/analysis combination should be used is driven in part by the known attributes of the device being tested and the kind of information being sought. For example, several swept sine measurements made at different stimulus levels can be used to characterize the operation of a device with an amplitude nonlinearity. Alternatively, an FFT measurement using random noise and averaging can be used to provide a single frequency response which approximates the device's operation over a range of stimulus levels.

If the device being tested is essentially linear (at least within the range of amplitudes and frequencies being tested), the selection of a stimulus/analysis combination is simply a matter of measurement speed. Any stimulus/analysis combination would be able to produce accurate results.

It is important to note, however, that before any assumption can be made about a system's linearity, at least two measurements (with variances in the stumuli between them) must be compared. If the system is found to be nonlinear, it may take several more measurements to characterize the nonlinearity so that its effect on the operation of the system can be understood.

It is in response to these measurement needs that advanced DSAs have incorporated the ability to make time domain measurements, traditional swept sine frequency response measurements and nontraditional frequency response measurements utilizing virtually any type of stimulus signal and FFT analysis. With these measurement capabilities, the DSA provides a total measurement solution for fully characterizing the operation of control systems.

Chapter 3: Analyze

Analyze: determining the value of parameters, either physical or theoretical, used to characterize the action or function of a device. Also, establishing the relationships, if any, between those parameters.

This definition of analysis, when applied to classical control theory, generally implies the evaluation of parameters such as gain margin, phase margin and settling time.

Typically, these parameters are not evaluated by the test instrument. More often than not, they must be derived from the measured data and, in some cases, derived from several sets of data. With respect to extracting useful information from measured data, the Dynamic Signal Analyzer represents one of the most powerful measurement and analysis tools available to the control systems engineer.

The DSA's major contributions toward analyzing data center around three major functions: waveform math, curve fitting and coherence. The following sections briefly describe each function and present typical applications.

3-1: Waveform Math

Waveform math provides the ability to use standard math operators such as +, −, × and ÷ between two displayed data traces, or perform any of the other math functions shown in Table 2-1 on individual traces. Waveform math therefore allows many of the control system calculations which have historically been done graphically, with plotted data, to be performed within the analyzer. This not only reduces calculation times, but also preserves the full resolution and accuracy of the original data. The following examples present only a few of the many possible applications for the waveform math function.

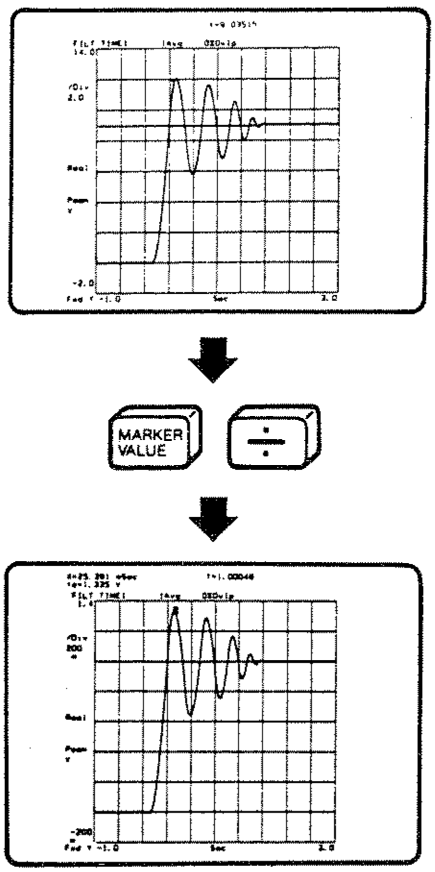
A very straightforward application of waveform math is the extraction of the normalized value of maximum overshoot from a step response measurement. The left half of Figure 2-5 shows a measured step response with a Y-axis marker positioned on the steady-state value. Using waveform math, the display can be normalized by simply specifying the ÷ operator and entering the response's steady-state value. The normalized value of maximum overshoot can then be read directly from the X-axis marker as shown in the right-half of Figure 2-5.

TABLE 2-1.

WAVEFORM MATH FUNCTIONS AVAILABLE WITH ADVANCED DYNAMIC SIGNAL ANALYZERS	+	$\sqrt{\ }$	MULTIPLY BY $j\omega$	T/(1−T)
	−	RECIPROCAL	MULTIPLY BY $j\omega$ −1	REAL PART
	×	NEGATE	FFT	COMPLEX CONJUGATE
	÷	DIFFERENTIATE	FFT−1	LOG OF DATA

FIGURE 2-5.

USING WAVEFORM MATH TO NORMALIZE A STEP RESPONSE. MARKER ON NORMALIZED DISPLAY READS OUT NORMALIZED PEAK OVERSHOOT DIRECTLY



Using the normalized display, the settling time can also be quickly evaluated. The upper and lower boundries relative to the steady-state value can be clearly marked by simply programming the Y-axis markers to those values (i.e., for a restriction of ±5% of final value, the markers can be set to 1.05 and 0.95). The X-axis marker can then be used to display the settling time, as shown in Figure 2-6. The information shown on the display of the DSA, including trace, display grid and annotation, can then be sent directly to a digital plotter to provide hardcopy documentation.

The DSA's waveform math function can also be used with frequency domain data to execute much more complex calculations. For example, two sets of frequency response data representing the forward gain path and feedback path of a system could be quickly combined to predict the system's open-loop frequency response.

Combining frequency responses can be accomplished by simply displaying one set of frequency response data in one display trace and a second set of frequency response data in the other display trace. The operator then selects an active trace, the multiply operator and the second operand (in this case the nonactive display trace)[1]. The result of the calculation is then displayed in the active trace, as shown in Figure 2-7.

FIGURE 2-6.

USING X- AND Y-AXIS MARKERS TO READ OUT SETTLING TIME OF A STEP RESPONSE FROM A NORMALIZED DISPLAY

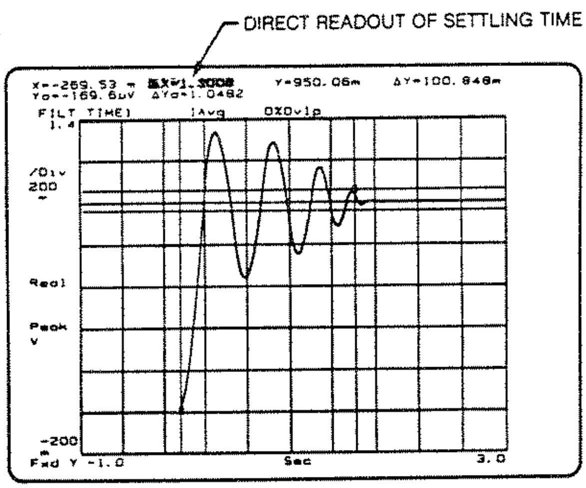
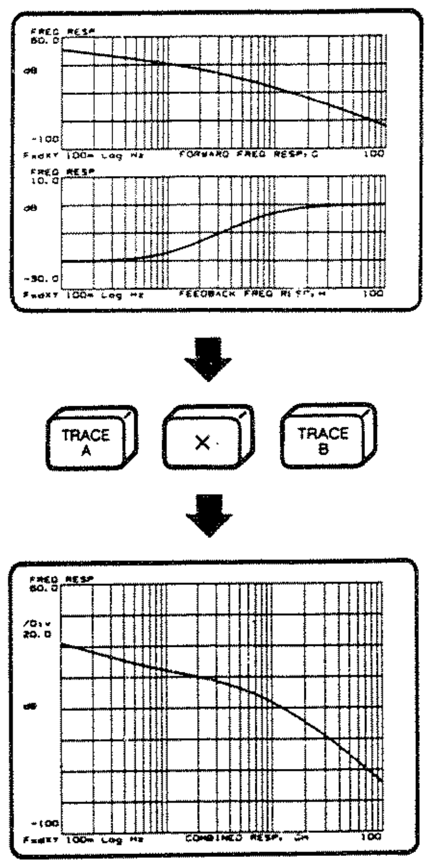


FIGURE 2-7.

USING WAVEFORM MATH TO COMBINE FREQUENCY RESPONSE DATA



[1] The order in which the waveform math operations are executed may differ between DSAs.

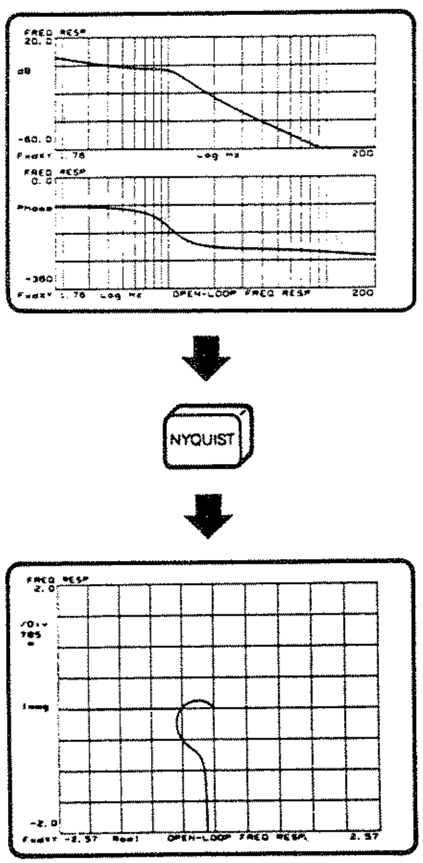
The resultant frequency response can then be presented in virtually any desired scale and in one of many display formats. For example, the derived frequency response can be displayed in a Bode plot, as shown in Figure 2-8a, to allow the gain margin, phase margin and open-loop bandwidth to be quickly read from the X-axis markers. The frequency response can then be displayed on a Nyquist plot, as shown in Figure 2-8b, to provide a quick check of the system's absolute stability.

Since either display trace may contain either current measurement data, calculated data, or data recalled from a mass storage device (such as a magnetic disc or tape drive), waveform math can be used to combine many frequency response data sets. This capability could be used to predict the frequency response of a system from a library of previously stored component frequency response data.

Waveform math also makes it possible to easily calculate the open-loop frequency response of a system from a closed-loop measurement. Typically, a stimulus signal is injected into the loop and, when using FFT analysis[1], the frequency response between the stimulus signal S and the response to the stimulus signal at the point Y is measured

FIGURE 2-8.

CRT DISPLAY OF DATA ALLOWS QUICK CHANGE OF DISPLAY FORMATS



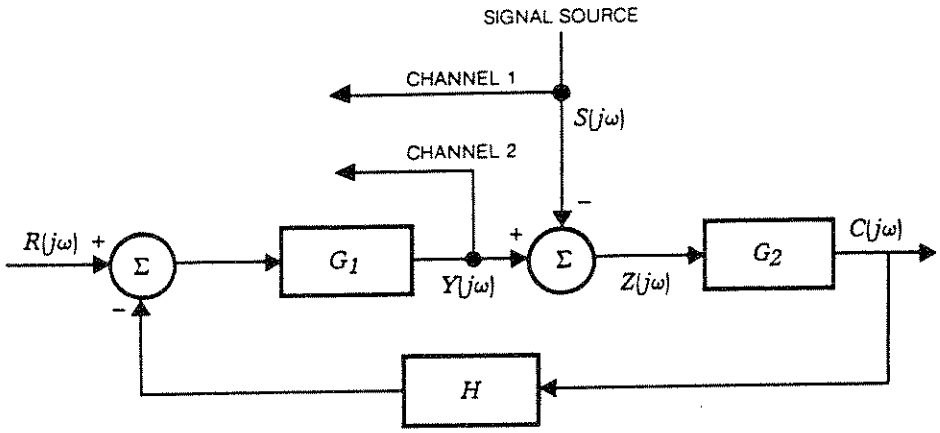
a. Bode Plot

b. Nyquist Plot

[1] When using an FFT analyzer to derive the open-loop frequency response of a closed-loop system, the ratio $Y(j\omega)/S(j\omega)$ or $Z(j\omega)/S(j\omega)$ is measured rather than $Y(j\omega)/Z(j\omega)$ (the ratio commonly measured with frequency response analyzers) to prevent a bias error from degrading the calculation. The bias error can be avoided and is typically not a significant factor when using SFA analysis.

FIGURE 2-9.

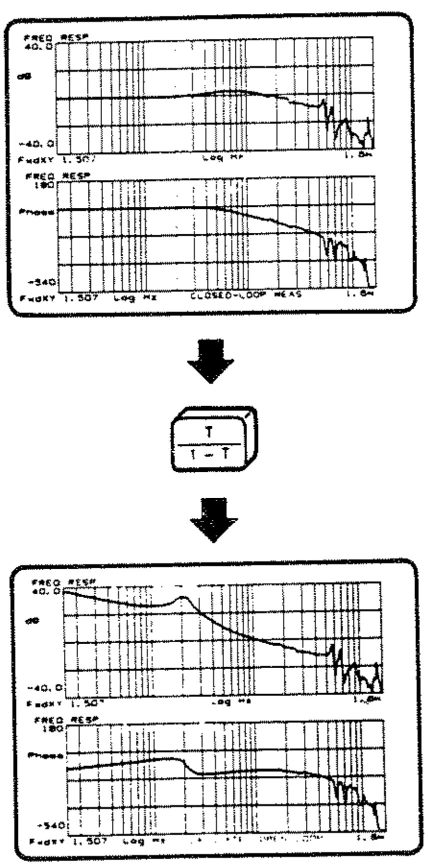
TYPICAL MEASUREMENT SETUP FOR DETERMINING THE OPEN-LOOP FREQUENCY RESPONSE OF A CLOSED-LOOP SYSTEM USING FFT



$$\frac{Y(j\omega)}{S(j\omega)} = \frac{G_1G_2H(j\omega)}{1 + G_1G_2H(j\omega)} = T(j\omega) \qquad \frac{T(j\omega)}{1 - T(j\omega)} = G_1G_2H(j\omega)$$

FIGURE 2-10.

USING THE "T/(1 − T)" WAVEFORM MATH FUNCTION TO CALCULATE THE OPEN-LOOP FREQUENCY RESPONSE FROM A MEASUREMENT OF $Y(j\omega)/S(j\omega)$



as shown in Figure 2-9. The open-loop frequency response of the system can then be calculated by evaluating the equation:

$$\text{open-loop frequency response} = \frac{T(j\omega)}{1 - T(j\omega)}$$

where $T(j\omega)$ is the measured frequency response $Y(j\omega)/S(j\omega)$.

This equation can be easily evaluated using either a series of waveform math calculations or by using the single waveform math operator T/(1−T) as shown in Figure 2-10.

3-2: Curve Fitting

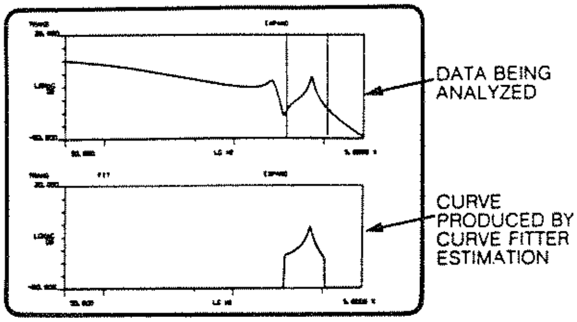
Curve fitting is a function which estimates an equation whose solution, when plotted, will be identical to the measured frequency response. Depending on the curve fitter available with a given DSA, the derived equation may be expressed to the operator in one of three formats: a table of poles and zeros, a table of poles and residues (i.e., partial fraction expansion form), or a ratio of polynomials.

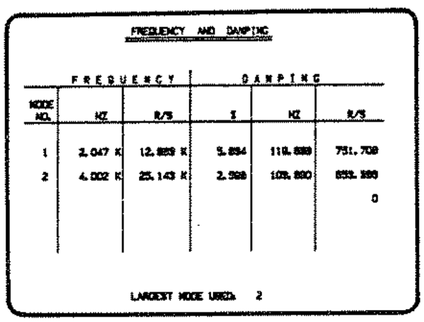
Advanced DSAs are usually equipped with one of two curve fitters, either a basic single-degree-of-freedom (SDOF) curve fitter or a multiple-degree-of-freedom (MDOF) curve fitter. SDOF curve fitters provide pole/residue information for each resonance identified by the operator, as shown in Figure 2-11a. MDOF curve fitters represent a more versatile generation of curve fitters which can automatically process an entire spectrum; using up to 40 poles and 40 zeros in the estimation process (see Figure 2-11b). The latter curve fitters are typically accompanied by a synthesis capability which allows the pole/zero information to be quickly converted to a pole/residue format or a polynomial format as shown in Figure 2-12.

For extracting information from measured data, the curve fitting function is an exceptionally powerful analysis tool. Its applications, however, lie mostly in the area of modeling and design and are discussed in chapters 4 and 5, in Part Two, respectively.

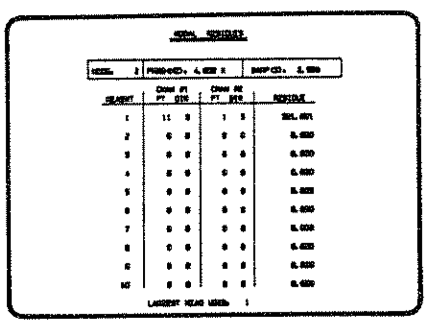
FIGURE 2-11.

USING CURVE FITTERS TO REDUCE FREQUENCY RESPONSE DATA TO ANALYTICAL EQUATIONS



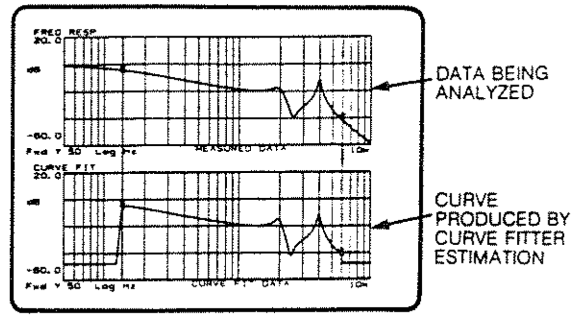


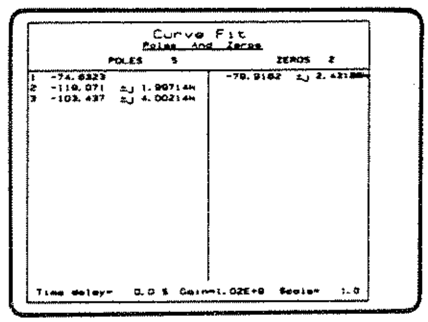
AND



$$\frac{R_1}{s+P_1} + \frac{R_1^*}{s+P_1^*} + \frac{R_2}{s+P_2} + \ldots = \text{TRANSFER FUNCTION}$$

a. Using an SDOF Curve Fitter to Evaluate the Frequency, Damping and Residue of Each Resonance in the Displayed Frequency Response. Curve Fit Data is Accumulated in the FREQUENCY AND DAMPING and MODAL RESIDUES Tables.





$$\frac{(s+Z_1)(s+Z_2)(s+Z_3)\ldots}{(s+P_1)(s+P_2)(s+P_3)\ldots} = \text{TRANSFER FUNCTION}$$

b. Using an MDOF Curve Fitter to Evaluate the Poles and Zeros Associated with the Displayed Frequency Response. Up to 40 Poles Can be Evaluated in One Analysis.

FIGURE 2-12.

CONVERTING POLE/ZERO DATA FROM THE CURVE FITTER TO POLE/RESIDUE AND POLYNOMIAL FORMATS VIA THE FREQUENCY RESPONSE SYNTHESIS TABLE CONVERSION FUNCTION

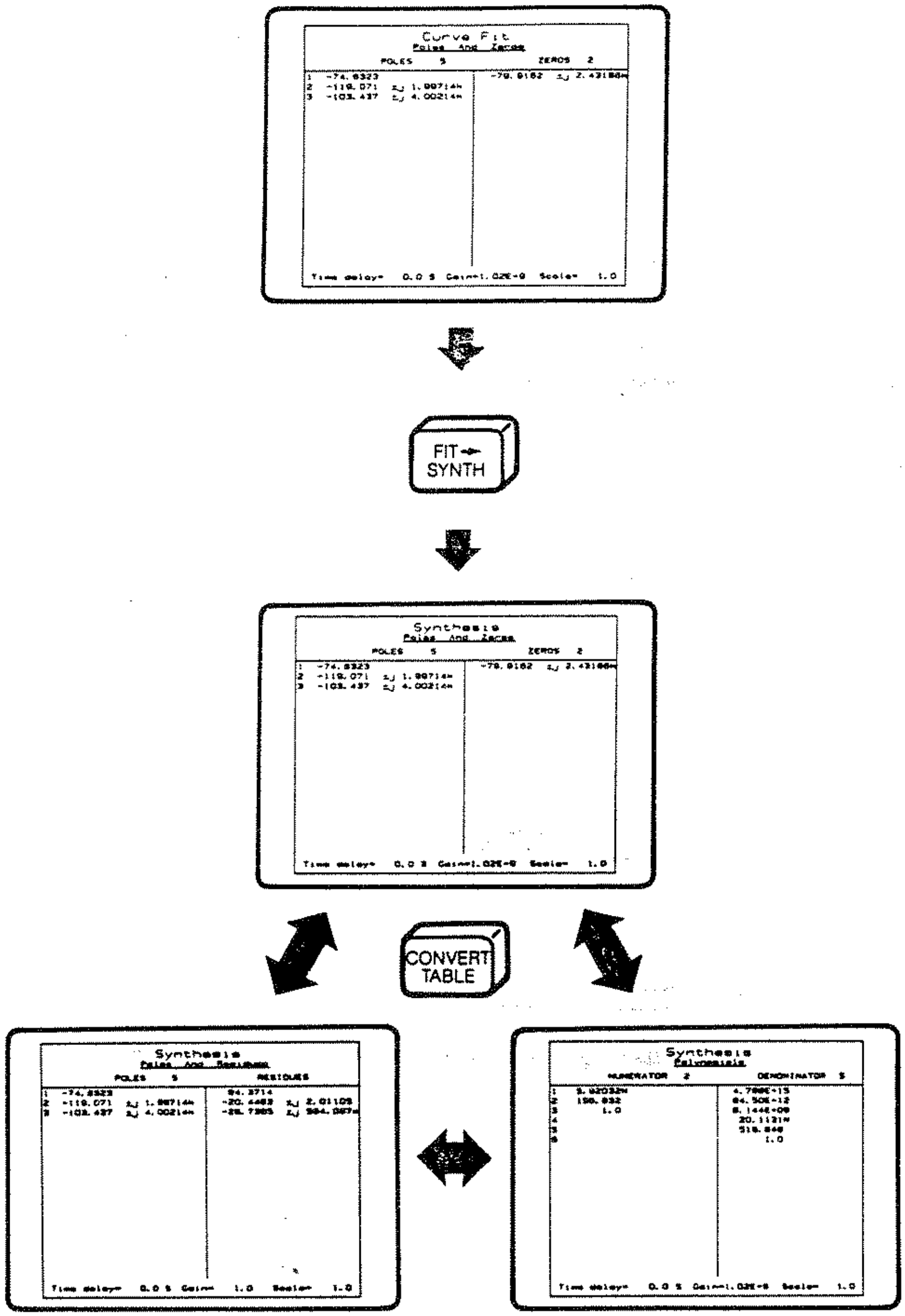
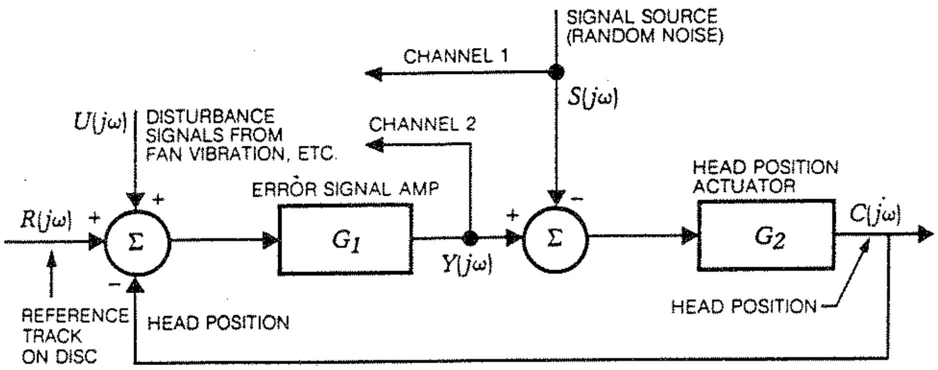
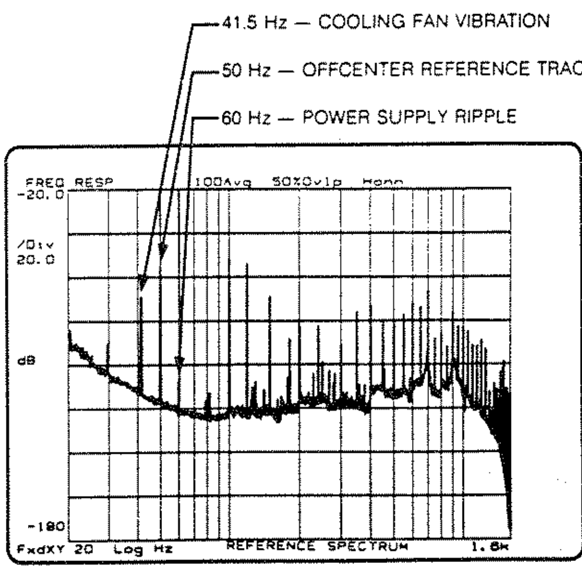


FIGURE 2-13.

USING THE NONCOHERENT POWER SPECTRUM OF THE RESPONSE TO MONITOR KNOWN PERIODIC DISTURBANCES IN THE CONTROL LOOP



a. Measurement Setup Used to Measure the Frequency Response $Y(j\omega)/S(j\omega)$ of a Disc Drive Read/Write Head Positioning Servo.



b. Noncoherent Power Spectrum Derived From $Y(j\omega)/S(j\omega)$ Measurement and the Associated Coherence Data. Note that the Error Signal Caused by an Offcenter Reference Track Appears as a Noncoherent Signal at 50 Hz.

3-3: Coherence

The coherence function is a statistical quantity whose dimensionless values represent the fraction of system output power directly related to the input. Values of coherence are used in two primary applications: 1) as a measure of the quality of a frequency response measurement and 2) to discriminate between those response signals which are directly related to (coherent with) the stimulus signal and those response signals which are not directly related to (not coherent with) the stimulus signal.

When more than one average is taken per measurement point, the coherence function produces a value from 0.0 to 1.0 for each point. (For example, when using SFA, a value of coherence will be produced for each step in the sweep if the analyzer is programmed to average two or more measurements per step.) A coherence value of 1 indicates that all of the output power (response) is coherent with the input power (stimulus) but not necessarily a result of the input power. A coherence value of 0 indicates that virtually none of the output power is coherent with the input power.

Since a low value of coherence indicates that only a small percentage of the response is directly related to the stimulus, it is reasonable to assume that the corresponding measurement data may not accurately reflect the transfer of energy through the tested device. In this respect, the coherence function acts as a qualitative tool which can be used to verify the general quality or credibility of a measurement. Typical causes of low coherence include very poor signal-to-noise ratios, the presence of noncoherent signals generated within the tested device or, when using FFT analysis, leakage due to improper window selection or insufficient time record length[1].

Coherence can also be used to separate the output power spectrum into two power spectra: the *coherent power spectrum* which represents the output power directly related to the input and the *noncoherent power spectrum* which represents the output power not related to the input.

Both the coherent and noncoherent power spectra have been used in several interesting applications. One example is the use of the noncoherent power spectrum by a disc drive manufacturer to monitor the disturbance signals within the read/write head positioning servo. By using a random stimulus signal, the periodic signals within the control loop (such as those caused by cooling fan vibration, power supply ripple bleeding into the control loop or an off-centered reference track on the disc) appear in the response as noncoherent signals. By correlating the known characteristic frequencies of these signals with the spectral components of the noncoherent power spectrum, the amplitudes of these noncoherent signals were effectively monitored, providing more information about the overall health of the positioning system. A simplified drawing of the measurement setup and an actual plot of the noncoherent power spectrum are shown in Figure 2-13.

[1] Complete definitions of leakage, window functions and time records are available in Hewlett-Packard Application Note 243, *The Fundamentals of Signal Analysis.*

Since all of the data needed to calculate the noncoherent power spectrum is provided with each frequency response measurement, it can be provided without increasing measurement time. The ability to increase the information obtained from each measurement can be especially valuable in situations where testing time is considered a valuable commodity, such as production line testing. A copy of a production test report dumped directly to a digital plotter by a DSA is shown in Figure 2-14.

The coherent and noncoherent power spectra mentioned above can easily be obtained by using waveform math to calculate the following formulas:

coherent power spectrum = (output power spectrum) × (coherence spectrum)

noncoherent power spectrum = (output power spectrum) × (1 − coherence spectrum)

where: coherence spectrum refers to the collective set of coherence values which exist when more than one average is taken and (1 − coherence spectrum) implies the subtraction of each value of coherence in the coherence spectrum from 1.

The output power spectrum, like the coherence function, is a normal by-product of a DSA's frequency response calculations and can be viewed at any time.

More applications for the coherence function (as well as a detailed definition) are provided in Hewlett-Packard Application Note 245-2, *Measuring the Coherence Function with the HP 3582A Spectrum Analyzer.*

Chapter 4: Model

Model: *the process of transforming the observed characteristics of some device or process into theoretical representations consistent with the analysis/design technique being used.*

This definition, when applied to classical control theory, generally implies the creation of equations which accurately predict the action or function of some device in the frequency or time domains. Since most design work is done in the frequency domain, the modeling process can further be generalized as the development of frequency domain equations, typically in a pole/zero format, which accurately predict a device's frequency response.

4-1: Curve Fitting Applied to the Modeling Process

As an aid in accomplishing this task, the MDOF curve fitter offered with high performance DSAs represents one of the most powerful tools ever offered by a test instrument.

By simply displaying a measured frequency response and activating the MDOF curve fitter, the DSA automatically provides an estimate of the *s*-plane poles and zeros and the gain required to produce the displayed response, as shown in Figure 2-15.

The use of a curve fitter to extract pole/zero information from a measured frequency response represents a significant advancement over the graphic techniques commonly used to derive pole/zero information. The curve fitter has the advantage of utilizing the full frequency and amplitude resolution of the measured data and, in many cases, provides the pole/zero information in the time normally required to obtain and prepare hardcopy plots for graphic interpretation.

FIGURE 2-14.

A PRODUCTION TEST REPORT SHOWING THE DATA PROVIDED FROM A MEASUREMENT MADE ON AN OPERATING CLOSED-LOOP SERVO SYSTEM

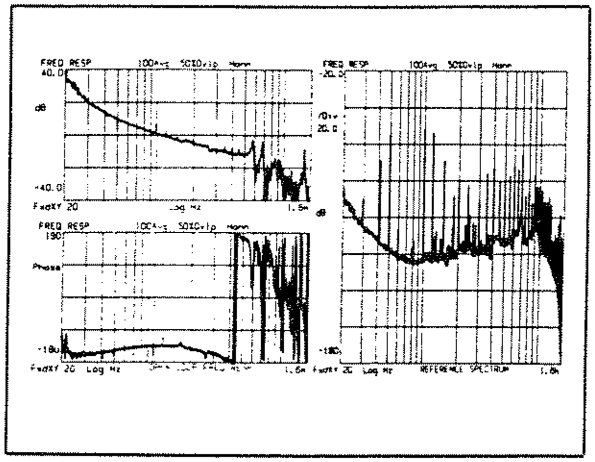


FIGURE 2-15.

USING AN MDOF CURVE FITTER TO ESTIMATE THE POLE/ZERO LOCATIONS AND GAIN FROM FREQUENCY RESPONSE DATA

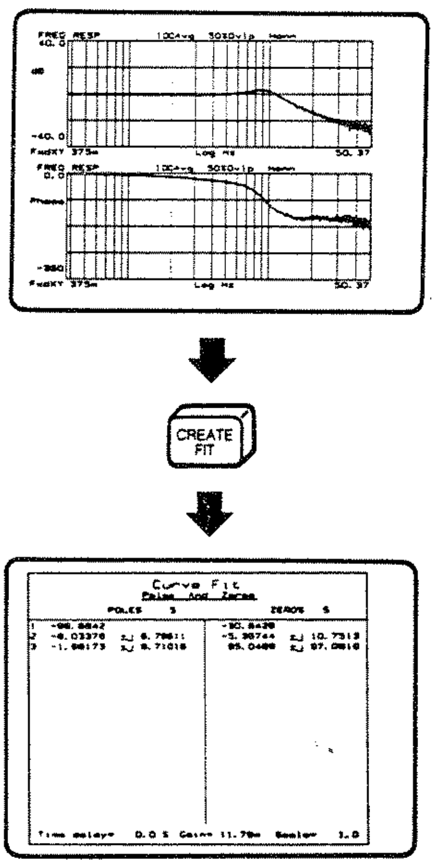
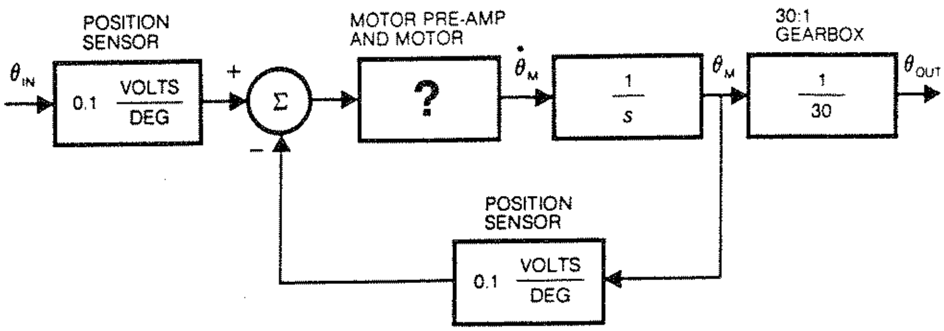
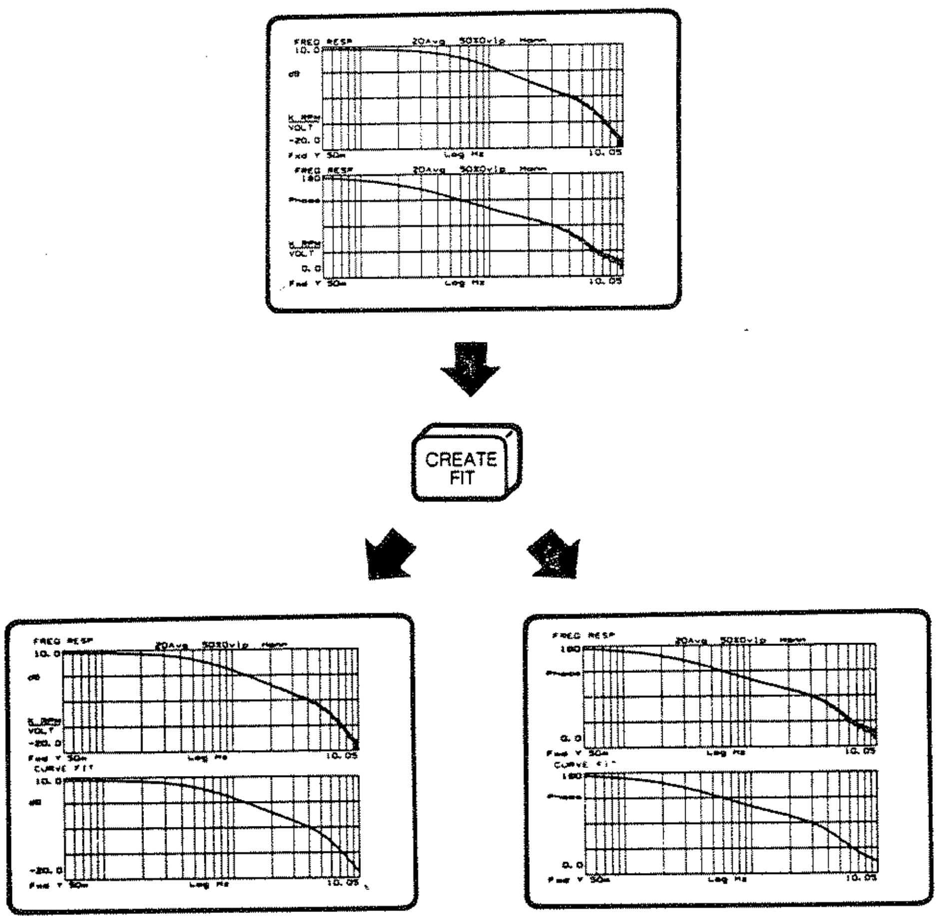


FIGURE 2-16.

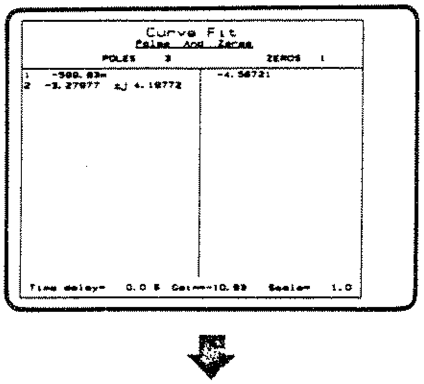


a. Block Diagram of Position Control System with Unknown Transfer Function for Motor and Pre-Amp.



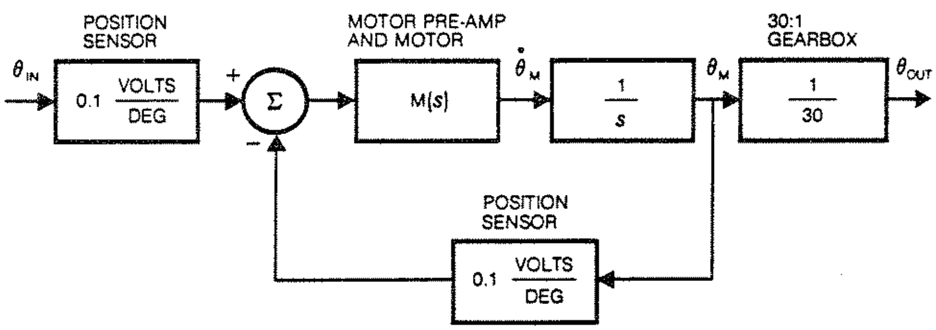
b. Curve Fitting the Measured Frequency Response of the Motor and Pre-amp to Produce an Estimate of the Associated Transfer Function (Poles, Zeros and Gain). The Upper/Lower Display Format is Used After the Fit to Compare the Measured Frequency Response with the Frequency Response of the Estimated Transfer Function.

FIGURE 2-16. (CONT.)



$$\frac{-10.83(s + 4.57)}{(s + 0.6)\,(s + 3.28 + j4.2)\,(s + 3.28 - j4.2)} = M(s)$$

c. Producing the Transfer Function of the Motor and the Pre-Amp from Pole/Zero Table Generated by DSA's MDOF Curve Fitter.



d. Completed Block Diagram.

Amongst other applications, the pole/zero data obtained from frequency response measurements can be used to either verify the poles and zeros used in an existing analytical model or create an initial model of a device with unknown characteristics. An example of the latter application is illustrated in Figure 2-16. In this example, a transfer function is generated for a combination armature controlled motor and pre-amplifier (of a position control system) whose specifications, such as motor inertia and forward gain, are unknown.

To obtain the motor/pre-amp's transfer function, the frequency response of the motor/pre-amp is first measured using a DSA equipped with a MDOF curve fitter. The curve fitter is then activated resulting in a table of poles and zeros. The pole/zero information is automatically synthesized to provide a frequency response which can be compared with the measured frequency response, as shown in Figure 2-16b. The pole/zero data is then used to generate a transfer function of the motor/pre-amp as illustrated in Figure 2-16c. The derived transfer function can now be added to the system block diagram to complete the system model, as shown in Figure 2-16d.

4-2: Frequency Response Synthesis Applied to the Modeling Process

Another useful modeling tool provided with advanced DSAs is the frequency response synthesis function (commonly referred to as the *synthesis* function). DSAs equipped with this function allow analytical equations (e.g., transfer functions) to be entered directly into the analyzer. The DSA then calculates and displays the frequency response associated with the transfer function, as shown in Figure 2-17.

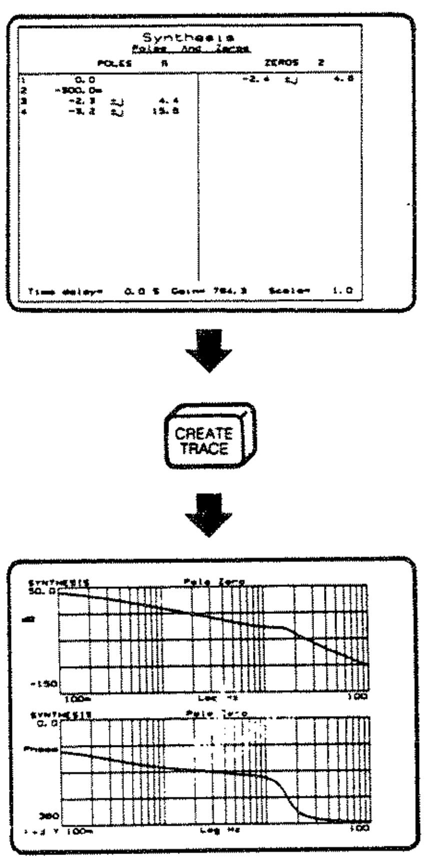
Equations may be entered in one of three formats: pole/zero, pole/residue (i.e., partial fraction expansion), or ratio of polynomials in *s*. In addition to providing a conversion function for transferring data from one format to another, high performance DSAs also provide direct transfer of data between the synthesis and curve fitting functions.

In the modeling process, the synthesis function is commonly used in conjunction with the curve fitter. For example, if the curve fitter produces more detailed information than required for a given application, the pole/zero data can be transferred to the synthesis function where insignificant poles and zeros can be deleted. The frequency response of the remaining poles and zeros can then be synthesized and compared to the measured frequency response. This allows the engineer to verify that the remaining poles and zeros sufficiently model the measured frequency response.

Another use of the synthesis function utilizes modeling information to optimize the initial testing of systems. By synthesizing the frequency response of a system which has never been tested (i.e. the model has been developed from data sheet information or initial design parameters), an initial estimate of the system's frequency response can be obtained. This information can then be used to estimate the transducers and stimulus levels required to properly test the system, reducing test time and, in many cases, preventing damage to the system or device being tested.

FIGURE 2-17.

SYNTHESIZING THE FREQUENCY RESPONSE OF A CONTROL SYSTEM USING THE DSA'S FREQUENCY RESPONSE SYNTHESIS FUNCTION



These examples illustrate only a few of the applications in which the DSA's precision measurement hardware and computational power contribute to the modeling process. By providing analysis tools such as frequency response synthesis and curve fitting, the DSA provides a new level of support for meeting the complex as well as the routine challenges of modeling today's control systems.

Chapter 5: Design

Design: determining the combination of physical or theoretical components or parameters that will produce a desired action or result.

The design process, as defined above, occurs throughout the development of control systems. It begins with the initial conception of a system and becomes one of an unpredictable sequence of development processes which ultimately result in a refined, fully operational control system. Typically, the purpose of most design work (after conceiving the initial system) is to generate modifications to the initial system which will allow it to comply with the original design goals or specifications. Modifications can range from simple changes in component values to the design and addition of complex compensation networks.

5-1: Applying Frequency Response Synthesis, Waveform Math and Curve Fitting to the Design Process

As a design tool, DSAs offer several data processing functions which can aid the engineer in choosing combinations of components which will accomplish a desired task. For example, the frequency response synthesis function[1] can be used to predict the frequency response of compensation networks before they are actually built. The waveform math function[2] can then be used to predict the effects of a synthesized compensation network on a system's open-loop frequency response or predict the system's new closed-loop frequency response. It can even be used to estimate the step or impulse response of the modified system before the compensation network is built.

To illustrate the use of the DSA's data processing functions in the design process, the following case study examines the development of a simple compensation network for a motor speed controller.

Initial measurements on the motor speed control were taken with the control loop closed and the system's open-loop gain set approximately 8 dB below the desired operating level. The closed-loop measurement indicated a sharp resonance at approximately 87.5 Hz, as shown in Figure 2-18a. The open-loop frequency response was then calculated from the measurement of $Y(j\omega)/S(j\omega)$ using the $T/(1-T)$ calculation, as shown in Figure 2-18b.

The magnitude of the resonance at 87.5 Hz indicated that an 8 dB increase in the gain would cause the open-loop gain at 90 Hz to exceed 0 db with the phase less than −180 degrees, creating an unstable operating condition, as shown in Figure 2-19. Therefore, to achieve the desired increase in the system's open-loop gain, a compensation network was added to the system to reduce the level of the 87.5 Hz resonance.

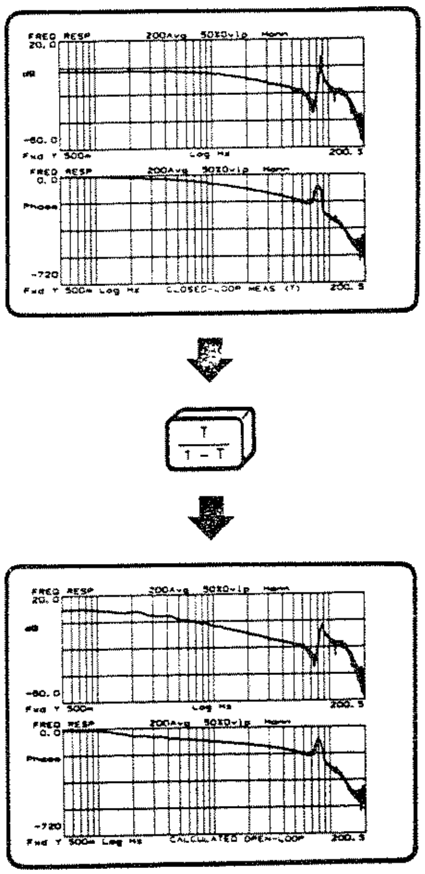
The compensation network, in this case a two-pole low-pass filter, was developed by entering an initial estimate of the pole locations, gain and delay into the pole/zero table of the DSA's frequency response synthesis function. The synthesized frequency response of the low-pass filter was then displayed on the CRT of the DSA, as shown in Figure 2-20.

[1] See section 4-2 for a brief description of the frequency response synthesis function.

[2] See section 3-1 for a brief description of the waveform math function.

FIGURE 2-18.

CLOSED-LOOP MEASUREMENT AND CALCULATED OPEN-LOOP FREQUENCY RESPONSE SHOWING RESONANCE AT 87.5 Hz

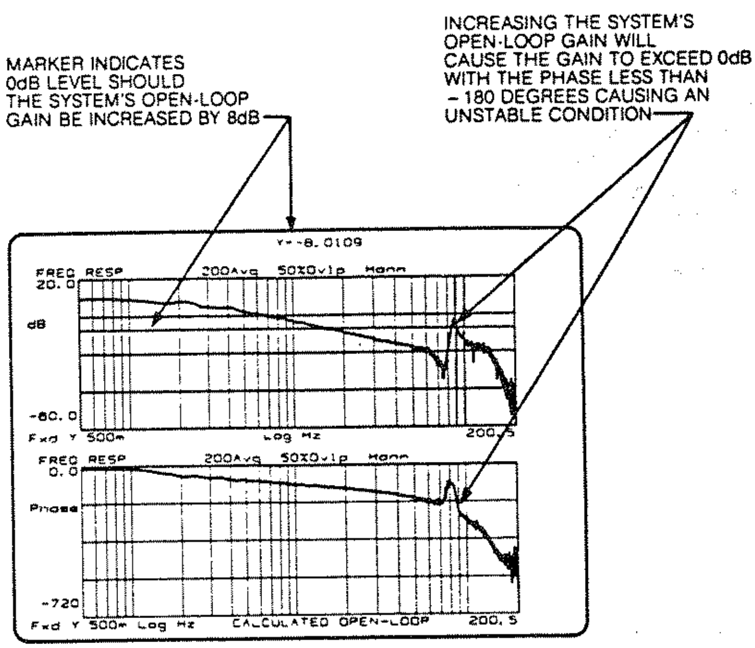


a. Closed-Loop Measurement
$(Y(j\omega)/S(j\omega))$

b. Calculated Open-Loop Frequency Response
$(GH(j\omega))$

FIGURE 2-19.

USING MARKERS TO PREDICT THE EFFECT OF INCREASING THE OPEN-LOOP GAIN BY 8 dB



The frequency response of the speed control system and the synthesized frequency response of the low-pass filter were then displayed adjacently, as shown in Figure 2-21. By displaying both frequency responses in this fashion, the low-pass filter pole locations which provided the best trade-off between level rejection and phase shift could quickly be determined.

FIGURE 2-20.

USING THE SYNTHESIS FUNCTION TO CALCULATE THE FREQUENCY RESPONSE OF A LOW PASS FILTER

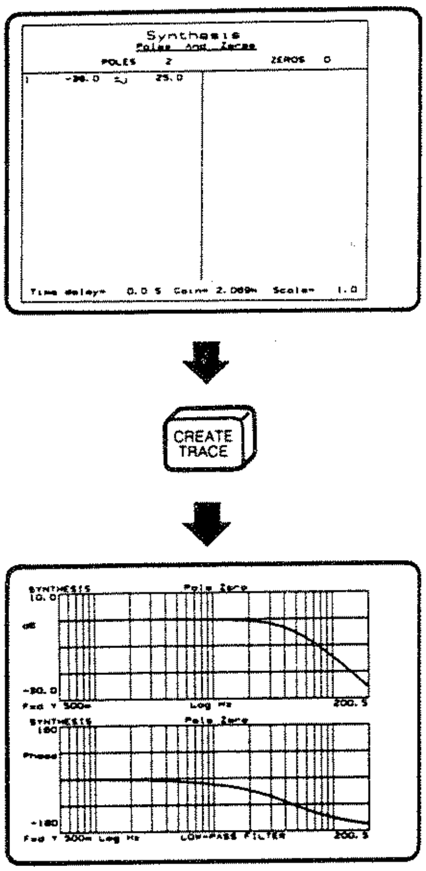
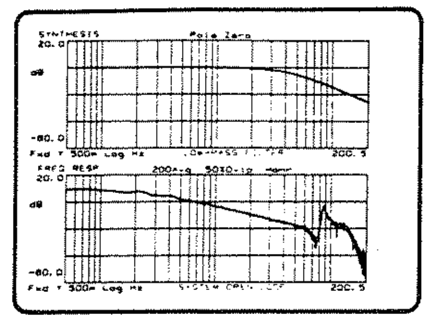
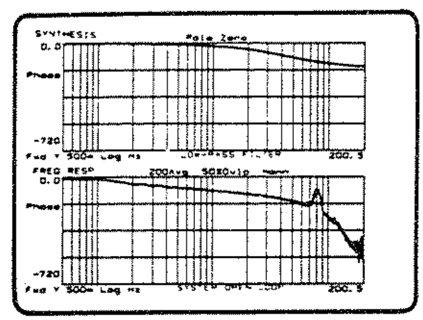


FIGURE 2-21.

SYNTHESIZED FREQUENCY RESPONSE OF LOW PASS FILTER (UPPER TRACE) AND CALCULATED OPEN-LOOP FREQUENCY RESPONSE OF MOTOR SPEED CONTROLLER (LOWER TRACE)



a. Gain



b. Phase

To verify the visual approximation, the synthesized frequency response of the low-pass filter was combined with the open-loop frequency response of the speed control system using waveform math, as shown in Figure 2-22.

Using the information provided by the pole/zero table and a passive filter design guide, the component values for the low-pass filter were determined and a prototype filter constructed. The frequency response of the prototype was then measured and compared to the synthesized frequency response, as shown in Figure 2-23.

FIGURE 2-22.

USING WAVEFORM MATH TO CALCULATE THE EFFECT OF THE LOW-PASS FILTER ON THE OPEN-LOOP FREQUENCY RESPONSE

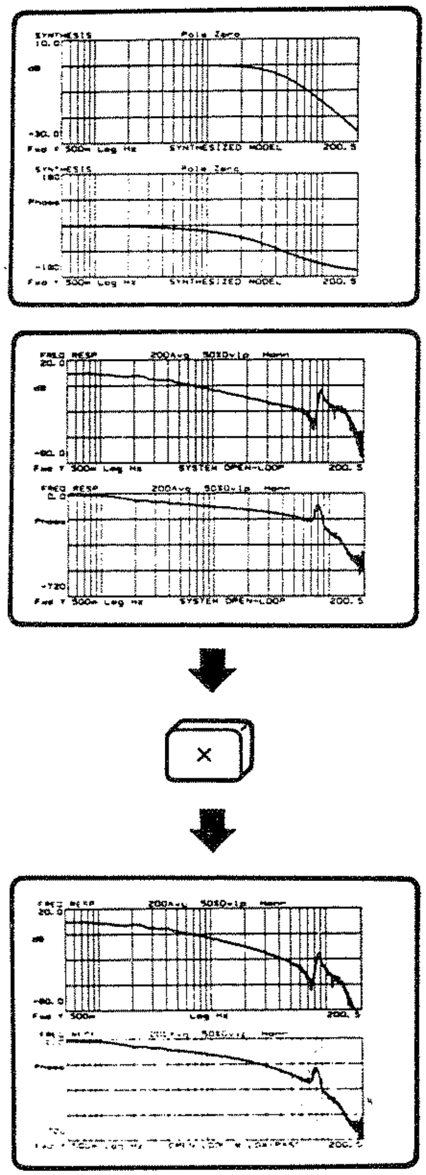
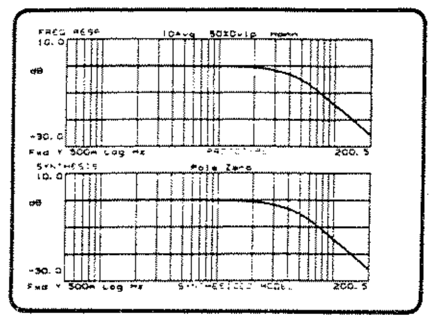
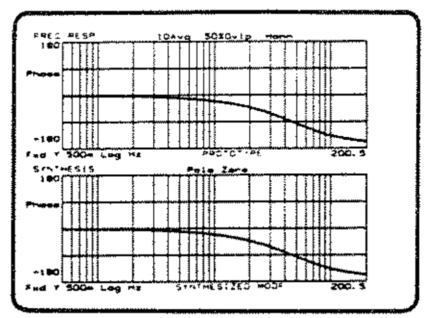


FIGURE 2-23.

MEASURED FREQUENCY RESPONSE OF PROTOTYPE (UPPER TRACE) COMPARED TO SYNTHESIZED FREQUENCY RESPONSE (LOWER TRACE)



a. Gain

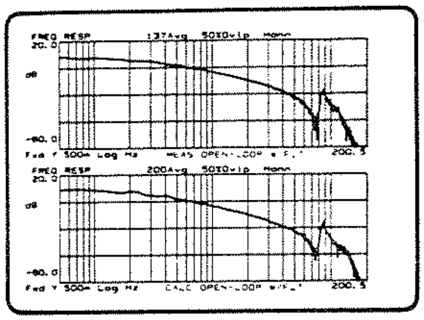


b. Phase

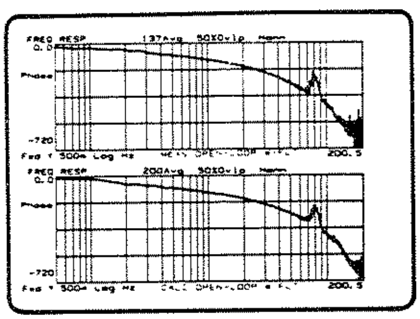
With the low-pass filter installed in the forward signal path of the motor speed control, the open-loop frequency response was again measured and compared to the predicted response, as shown in Figure 2-24. Finally, the gain of the speed control was raised by 8 dB to provide the desired performance while maintaining reasonable gain margin and phase margin, as shown in Figure 2-25.

FIGURE 2-24.

COMPARISON OF THE MEASURED (UPPER TRACE) AND THE PREDICTED (LOWER TRACE) OPEN-LOOP FREQUENCY RESPONSE WITH LOW-PASS FILTER INSTALLED



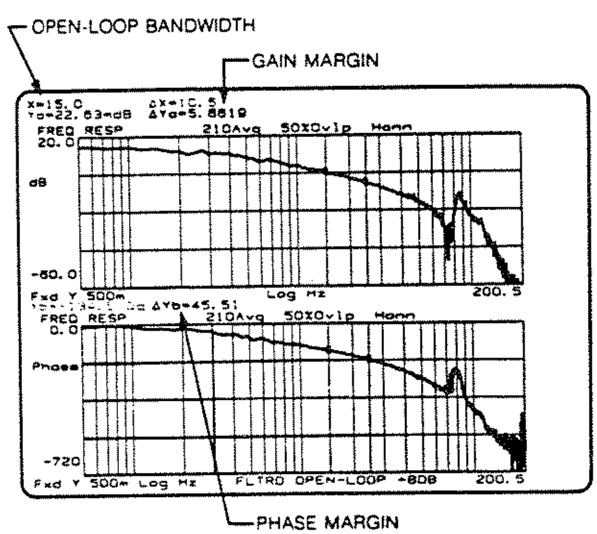
a. Gain



b. Phase

FIGURE 2-25.

MEASURED OPEN-LOOP FREQUENCY RESPONSE WITH LOW-PASS FILTER INSTALLED AND GAIN INCREASED BY 8 dB



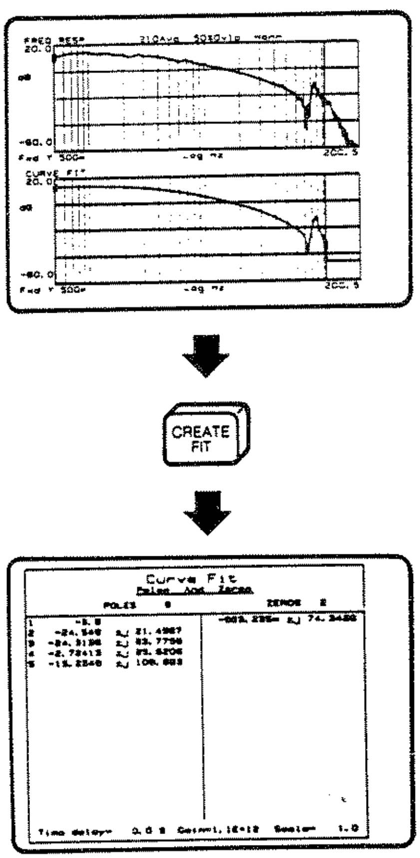
In this example, the low-pass filter provided enough compensation to achieve the desired system performance. However, for more demanding applications, a lag-lead network could be added to the system to further improve the system's performance.

When building compensation networks such as the lag-lead network mentioned above, the DSA's curve fitter can be used to locate the dominant poles and zeros of a system's open-loop frequency response, as shown in Figure 2-26. This information can then be used with design tools such as a root locus plot to select the most advantageous position for the poles and zeros of the compensation network.

The DSA's curve fitter function can also be used to suggest the location of a compensation network's poles and zeros. For example, the pole/zero model of a "perfect" compensation network can be derived using a combination of the frequency response synthesis, waveform math and curve fitting functions. First, the frequency response synthesis function is used to synthesize the "ideal" frequency response for a system. Waveform math is then used to divide the synthesized response by the system's measured frequency response. The result is the frequency response of the cascade compensation network needed to achieve the "ideal" frequency response for the system. By curve fitting this resultant frequency response, the DSA supplies the designer with a table of poles and zeros which will produce that response.

FIGURE 2-26.

CURVE FITTING OVER FREQUENCY RANGE OF INTEREST TO DETERMINE THE DOMINANT POLES AND ZEROS OF THE SYSTEM



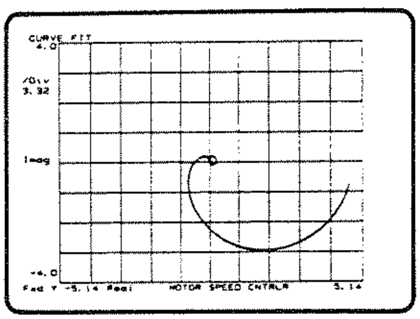
5-2: Using Display Formats Other Than the Bode Plot

By providing a wide choice of coordinate formats, advanced DSAs allow the operator to observe frequency response data in the display format which best conforms with the design technique being used. For example, the open-loop frequency response of the motor speed controller can be displayed in either the Nichols or Nyquist formats as shown in Figure 2-27.

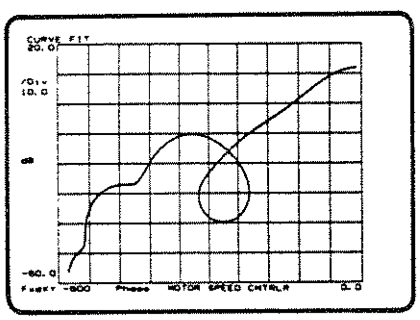
This rapid exchange of data between display formats not only allows the engineer to capitalize on the advantages of each display format, it also serves as a convenient way to bridge communication gaps between engineers accustomed to different display formats.

FIGURE 2-27.

OPEN-LOOP FREQUENCY RESPONSE OF MOTOR SPEED CONTROLLER REDISPLAYED IN THE NICHOLS AND NYQUIST FORMATS



a. Nyquist Plot of Open-Loop Frequency Response



b. Nichols Plot of Open-Loop Frequency Response

Chapter 6: Summary

By combining the computational power of the microprocessor with the accuracy of precision measurement hardware, the Dynamic Signal Analyzer has expanded its functional scope to include contributions in virtually all aspects of control system development.

In the area of testing, the DSA has provided the facilites for making both time domain and frequency domain measurements. Using either the time capture or time throughput measurement modes, the DSA can store large quantities of time domain data. The data can then be either displayed in the time domain or routed to the FFT processor and transformed into frequency domain data.

For making frequency domain measurements, the DSA provides both FFT analysis and Swept Fourier Analysis. This combination of measurement capabilities allows the DSA to analyze a control system's response to a wide range of stimulus signals. This capability can often be used to gain greater insight into the operation of a control system as well as minimize measurement times.

In addition to providing multiple measurement capabilities, the DSA utilizes the power of the microprocessor to provide a host of automated measurement aids capable of optimizing measurement conditions and rejecting undesirable data.

In the area of analysis, the DSA provides functions such as coherence, waveform math, curve fitting and advanced display formatting as tools for reducing raw data to valuable information.

In the areas of modeling and design, the DSA's frequency response synthesis and advanced analysis functions can be utilized in the development of accurate system models and effective system designs.

Perhaps the DSA's most significant contribution is that it has brought both advanced measurement capabilities and powerful analysis tools together in a single instrument. This consolidation of development tools allows the DSA to provide a great deal of valuable information—not just data.

Appendix A: Glossary

Bandwidth. The interval separating two frequencies between which both the gain and the phase difference (of sinusoidal output referred to sinusoidal input) remain within specified limits.
Bode diagram. A plot of log-gain and phase-angle values on a log-frequency base, for an element transfer function $G(j\omega)$, a loop transfer function $GH(j\omega)$. The generalized Bode diagram comprises similar plots of functions of the complex variable $s = \sigma + j\omega$.
Characteristic equation. Of a feedback control system, the relation formed by equating to zero the denominator of a rationalized transfer function of a closed loop.
Closed loop (feedback loop). A signal path which includes a forward path, a feedback path and a summing point, and forms a closed circuit.
Compensation. A modifying or supplementary action (also, the effect of such action) intended to improve performance with respect to some specified characteristic.
Control system. A system in which deliberate guidance or manipulation is used to achieve a prescribed value of a variable.
NOTE: It may be subdivided into a controlling system and a controlled system.
Control system, automatic. A control system which operates without human intervention.
Control system, feedback. A control system which operates to achieve prescribed relationships between selected system variables by comparing functions of these variables and using the difference to effect control.
Control system, open-loop. One which does not utilize feedback of measured variables.
Critically damped. Describing a linear second-order system which is damped just enough to prevent any overshoot of the output following an abrupt stimulus. See also damping.
Critical point. (1) In a Nyquist diagram for a control system, the bound of stability for the locus of the loop transfer function $GH(j\omega)$, the $(-1, j\sigma)$point. (2) In a Nichols chart, the bound of stability for the $GH(j\omega)$ plot; the intersection of $|GH| = 1$ with $\angle GH = -180$ degrees.
Damping. (1) (noun) The progressive reduction or supression of the oscillation of a system. (2) (adj.) Pertaining to or productive of damping.
Decibel. In control usage, a logarithmic scale unit relating a variable x (e.g., angular; displacement) to a specified reference level x_0; dB = 20 log x/x_0.
NOTE: The relation is strictly applicable only where the ratio x/x_0 is the square root of the power ratio P/P_0, as is true for voltage or current ratios. The value dB = 10 log P/P_0 originated in telephone engineering, and is approximately equivalent to the old "transmission unit".
Dither. A useful oscillation of small amplitude introduced to overcome the effects of friction, hysteresis or clogging.
Error constant. In a feedback control system, the real number K by which the nth derivative of the reference input signal is divided to give the resulting nth component of the actuating signal.
Frequency, damped. The apparent frequency of a damped oscillatory time response of a system resulting from a non-oscillatory stimulus.
Frequency, gain crossover. On a Bode diagram of the loop transfer function of a system, the frequency at which the gain becomes unity (and its decibel value zero).
Frequency, phase crossover. Of a loop transfer function the frequency at which the phase angle reaches ± 180 degrees.
Frequency response. In a linear system, the frequency-dependent relation in both gain and phase difference, between steady-state sinusoidal inputs and the resulting steady-state sinusoidal outputs.
Function describing. Of a nonlinear element under periodic input, a transfer function based solely on the fundamental, ignoring other frequencies.
Function, loop transfer. For a closed loop, the transfer function obtained by taking the ratio of the Laplace transform of the return signal to the Laplace transform of its corresponding error signal.
Function, output transfer. For a closed loop, the transfer function obtained by taking the ratio of the Laplace transform of the output signal to the Laplace transform of the input signal.
Function, return transfer. For a closed loop, the transfer function obtained by taking the ratio of the Laplace transform of the return signal to the Laplace transform of its corresponding input signal.
Function, system transfer. The transfer function obtained by taking the ratio of the Laplace transform of the signal corresponding to the ultimately controlled variable to the Laplace transform of the signal corresponding to the command.
Function, transfer. A mathematical, graphical, or tabular statement of the influence which a system or element has on a signal or action compared at input and at output terminals.
Gain (magnitude ratio). For a linear system or element, the ratio of the magnitude (amplitude) of a steady-state sinusoidal output relative to the causal input; the length of a phasor from the origin to a point of the transfer locus in a complex plane.
NOTE: The quantity may be separated into two factors: (1) a proportional amplification often denoted as K which is frequency-independent, and associated with a dimensioned scale factor relating the units of input and output; (2) a dimensionless factor often denoted as $G(j\omega)$ which is frequency-dependent. Frequency, conditions of operation, and conditions of measurement must be specified A loop gain characteristic is a plot of log gain vs. log frequency. In nonlinear systems, gains are often amplitude-dependent; see also transfer function.
Gain characteristic, loop. Of a closed loop, the magnitude of the loop transfer function for real frequencies.
Gain, closed-loop. The gain of a closed-loop system, expressed as the ratio of output to input.
Gain, loop. The absolute magnitude of the loop gain characteristic at a specified frequency.
Gain margin. Of the loop transfer function for a stable feedback system, the reciprocal of the gain at the frequency at which the phase angle reaches minus 180 degrees.
NOTE: Gain margin, sometimes expressed in decibels is a convenient way of estimating relative stability by Nyquist, Bode, or Nichols diagrams, for systems with similar gain and phase characteristics. In a conditionally stable feedback system, gain margin is understood to refer to the highest frequency at which the phase angle is minus 180 degrees.
M-peak. Of a closed loop, the maximum value of the magnitude of the return transfer function for real frequencies, the value at zero frequency being normalized to unity.
Nichols chart (Nichols diagram). A plot showing magnitude contours and phase contours of the return transfer function referred to ordinates of logarithmic loop gain and to abscissae of loop phase angle.

Nyquist diagram. A polar plot of the loop transfer function.
NOTE: The "inverse Nyquist diagram" is a polar plot of the reciprocal function. The generalized Nyquist diagram comprises plots of the loop transfer function of the complex variables, where $s = \sigma + j\omega$ and σ and ω are arbitrary constants, including zero.
Overdamped. Damped sufficiently to prevent any oscillation of the output following a step or impulse input.
NOTE: For a linear second-order system the roots of the characteristic equation are real and unequal.
Phase angle, loop. Of a closed loop, the value of the loop phase characteristic at a specified frequency.
Phase characteristic, loop. Of a closed loop, the phase angle of the loop transfer function for real frequencies.
Phase margin. Of the loop transfer function for a stable feedback control system, 180 deg. minus the absolute value of the loop phase angle at a frequency where the loop gain is unity.
NOTE: Phase margin is a convenient way of expressing relative stability of a linear system under parameter changes in Nyquist, Bode or Nichols diagrams. In a conditionally stable feedback control system where the loop gain becomes unity at several frequencies, the term is understood to apply to the value of phase margin at the highest of these frequencies.
Pole. (1) Of a transfer function in the complex variable s, a value of s which makes the function infinite. (2) The corresponding point in the s-plane.
NOTE: If the same value is repeated n times, it is called a pole of nth order; if it occurs only once, a simple pole.
Resonance. Of a system or element, a condition evidenced by large oscillatory amplitude which results when a small amplitude of a periodic input has a frequency approaching one of the natural frequencies of the driven system.
NOTE: In a feedback control system, this occurs near the stability limit.
Response, steady-state. Of a stable system or element, that part of the time response remaining after transients have expired.
NOTE: The term steady-state may also be applied to any of the forced response terms: for example, "steady-state sine-forced response".
Root locus. For a closed loop whose characteristic equation is $KG(s)H(s) + 1 = 0$, a plot in the s-plane of all those values of s which make $G(s)H(s)$ a negative real number; those points which make the loop transfer function $KG(s)H(s) = -1$ are roots.
NOTE: The locus is conveniently sketched from the factored form of $KG(s)H(s)$; each branch starts at a pole of that function, with $K = 0$. With increasing K, the locus proceeds along its several branches toward a zero of that function and, often asymptotic to one of several equi-angular radial lines, toward infinity. Roots lie at points on the locus for which (1) the sum of the phase angles of component $G(s)H(s)$ vectors totals 180 deg., and for which (2) $1/K = |G(s)H(s)|$. Critical damping of the closed loop occurs when the locus breaks away from the real axis; instability when it crosses the imaginary axis.
Servomechanism. An automatic feedback control system in which the controlled variable is mechanical position or any of its time derivatives.
Servomechanism type number. In control systems in which the loop transfer function is:

$$\frac{K(1 + a_1 s + a_2 s^2 + \ldots + a_i s^i)}{S^n(1 + b_1 s + b_2 s^2 + \ldots + b_k s^k)}$$

where K, a, b etc. are constant coefficients, the value of the integer n.
Stability. For a control system, the property that sufficiently bounded input or initial state perturbation result in bounded state or output perturbations.
Time, rise. The time required for the output of a system (other than first-order) to make the change from a small specified percentage (often 5 or 10) of the steady-state increment to a large specified percentage (often 90 or 95), either before overshoot or in the absence of overshoot.
NOTE: If the term is unqualified, response to a unit-step stimulus is understood, otherwise the pattern and magnitude of the stimulus should be specified.
Time, settling (correction time). The time required following the initiation of a specified stimulus to a linear system for the output to enter and remain within a specified narrow band centered on its steady-state value.
NOTE: The stimulus may be a step, impulse, ramp, parabola, or sinusoid. For a step or impulse, the band is often specified as ± 2%. For nonlinear behavior, both magnitude and pattern of the stimulus should be specified.
Underdamped. Damped insufficiently to prevent oscillation of the output following an abrupt stimulus.
Zero. (1) of a transfer function in the complex variable s, a value of s which makes the function zero. (2) The corresponding point in the s-plane.
NOTE: If the same value is repeated n times, it is called a zero of nth order; if it occurs only once, a simple zero.

Appendix B: **Bibliography**

American National Standard Association, 1981. *Terminology for Automatic Control,* ANSI MC85.1M-1981. American Society of Mechanical Engineers, New York, NY.

Brogan, W. L., 1974. *Modern Control Theory.* Quantum, New York, NY.

D'Azzo, J. J., and C. H. Houpis, 1981. *Feedback Control System Analysis and Synthesis.* McGraw-Hill, New York, NY.

DiStefano, J. J., A. R. Stubberud, and I. J. Williams, 1967. *Schaum's Outline of Theory and Problems of Feedback and Control Systems.* Schaum, New York, NY.

Dorf, R. C., 1980. *Modern Control Systems.* Addison-Wesley, Reading, MA.

Gabel, R. A., and R. A. Roberts. *Signals and Linear Systems.* John Wiley, New York, NY.

Hewlett-Packard Co. *Digital Signal Analysis, Feedback Control System Measurements.* Application note 240-1.

Hewlett-Packard Co. *Measuring the Coherence Function with the HP 3582A Spectrum Analyzer.* Application note 245-2.

Hewlett-Packard Co., 1982. *The Fundamentals of Signal Analysis.* Application note 243.

Institute of Electrical and Electronics Engineers. Standards Committee, 1972. *IEEE Standard Dictionary of Electrical and Electronics Terms.* IEEE, New York, NY, IEEE Standard 100-1972.

Kuo, B. C., 4th Ed., 1982. *Automatic Control Systems.* Prentice-Hall, Englewood Cliffs, NJ.

MacFarlane, A. G. J., ed. *Frequency-Response Methods in Control Systems.* IEEE Press, New York, NY.

Nagrath, I. J., and M. Gopal, 1982. *Control Systems Engineering.* Wiley Eastern, New Delhi, India.

Thaler, G. J., ed., 1974. *Automatic Control: Classical Linear Theory.* Dowden, Hutchinson, & Ross, Stroudsburg, PA.

Thaler, G. J. and R. G. Brown, 1960. *Analysis and Design of Feedback Control Systems.* McGraw-Hill, New York, NY.

Chen, C.T., 1984. *Linear System Theory and Design.* Holt, Rinehart, and Winston.

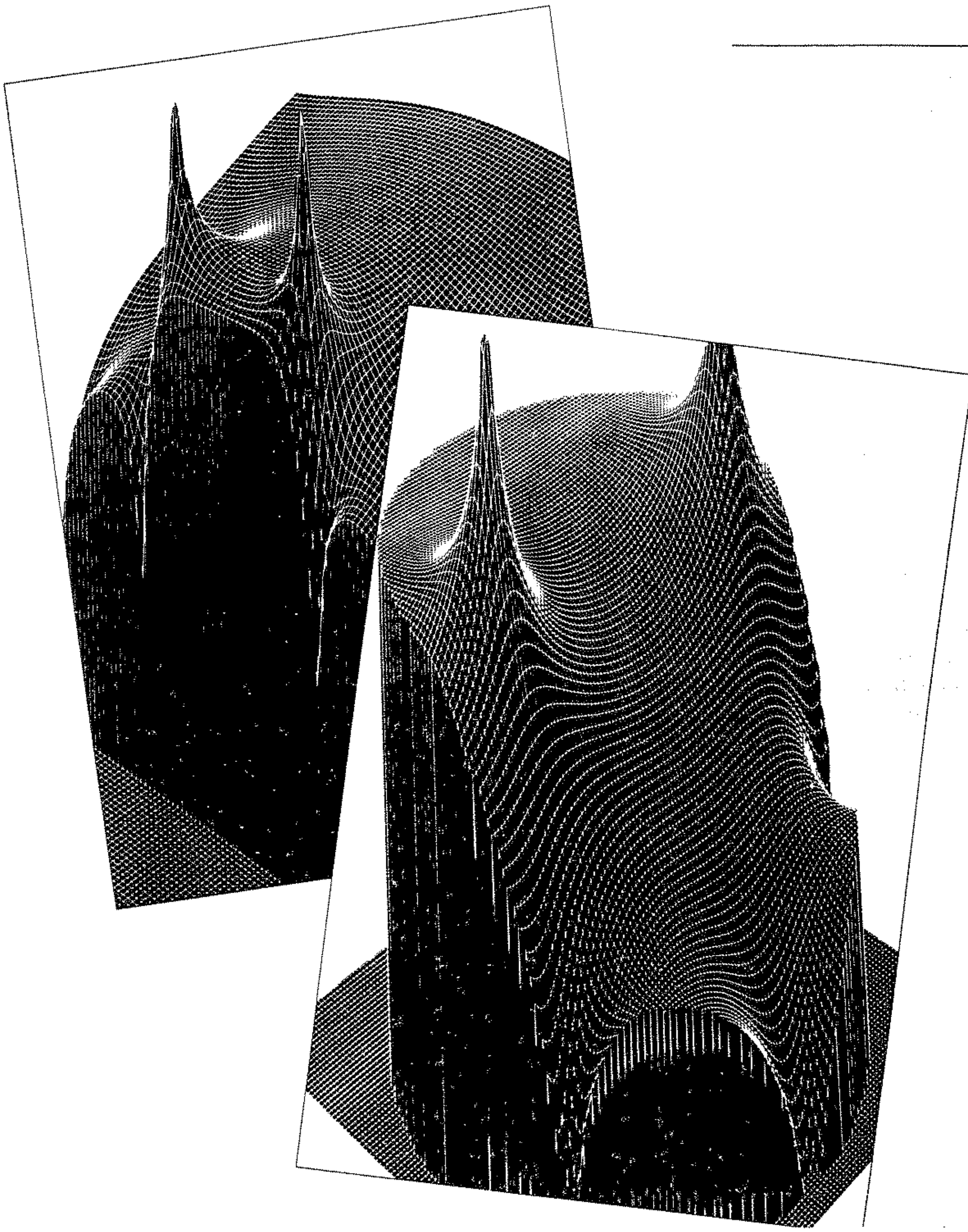
Appendix C: **Acknowledgements**

Many of the definitions within this application note have been either influenced by, or directly reprinted from American National Standard ANSI MC85.1M-1981 with permission from the publisher.

HEWLETT PACKARD

Fundamentals of the z-Domain and Mixed Analog/Digital Measurements

Application Note 243-4



Preface

There are numerous applications involving mixed analog and digital signals in the same system. In order to make measurements on mixed systems of this sort, it is helpful to use the z-transform for the digital part in conjunction with the Laplace transform (s-domain) for the analog part. In this note, the z-transform is defined and various transformations between the s and z domains are discussed. The Appendix is devoted to a discussion of matching the impulse responses of multiple poles in both the s and z domains.

The key characteristics of mixed domain measurements are also discussed in this note. For example, multiple images occur in the spectrum of a sampled signal. To measure the higher order images of the digital transfer function with a dynamic signal analyzer, the analog sampling rate is generally some integer multiple of the digital rate. To accurately measure the frequency response of a mixed system, these two sampling rates must be carefully locked together in both frequency and phase. There is also the need to handle time delays, both in the signal path and in the sampling pulse path.

Contents

Introduction

There are many applications in which signals are represented in both analog and digital form at different nodes in a system. For instance, control systems in which some part of the control loop is implemented in digital form, such as the loop compensation, are becoming more common. Figure 1 shows a simple block diagram of a mixed domain system which contains analog filters, data converters, and a digital filter. Measurements in mixed systems of this sort are somewhat more complicated than those for strictly analog systems since at least one of the time waveforms is only available in sampled form.

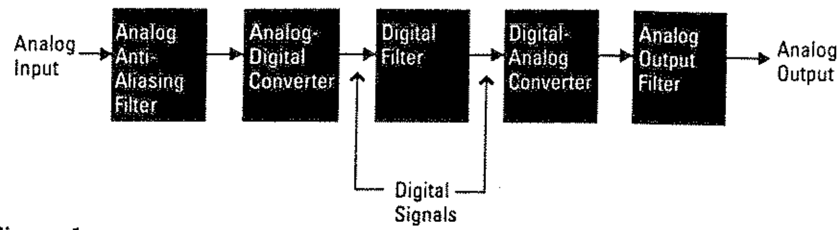


Figure 1: Block diagram for a mixed analog/ digital system.

The frequency response function of a sampled data system is periodic along the frequency axis, with images spaced at multiples of the sampling rate (see figure 8 for an example showing four images). This implies that poles and zeros in the original s-domain are also replicated along the frequency axis, resulting in an infinity of new poles and zeros at multiples of the sampling frequency. The analog part of a mixed domain system is generally designed to suppress frequencies corresponding to these higher order images in the digital domain. This is the purpose of the analog anti-aliasing and output filters in figure 1.

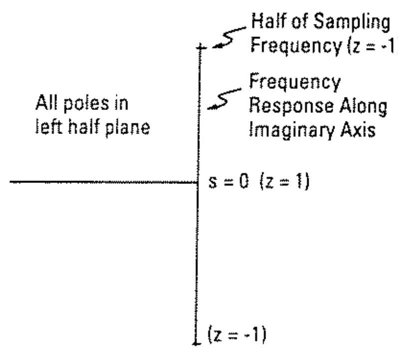


Figure 2: The s-domain, showing the frequency (imaginary) axis.

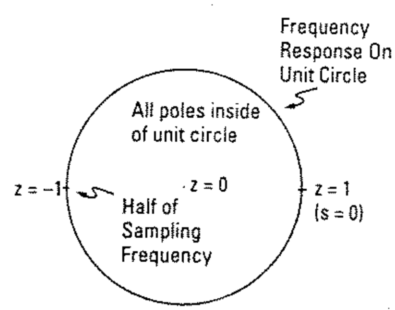


Figure 3: The z-domain, showing the unit circle that corresponds to the frequency axis in the s-plane.

The z-transform is used to characterize the transfer function of a sampled data system. This transform will be derived later, but it is simply a technique for representing a periodic frequency response function around a circle instead of along the linear frequency axis in the s-plane. Each frequency image is mapped onto one cycle around the unit circle. The values of the z-transform around the unit circle correspond to the measured frequency response function (at least below half of the sampling frequency), just as the values of the Laplace transform along the imaginary axis correspond to the measured frequency response function. All poles and zeros in the left half of the s-plane map into the interior of the unit circle in the z-plane, and the entire right half of the s-plane maps into the exterior of the unit circle.

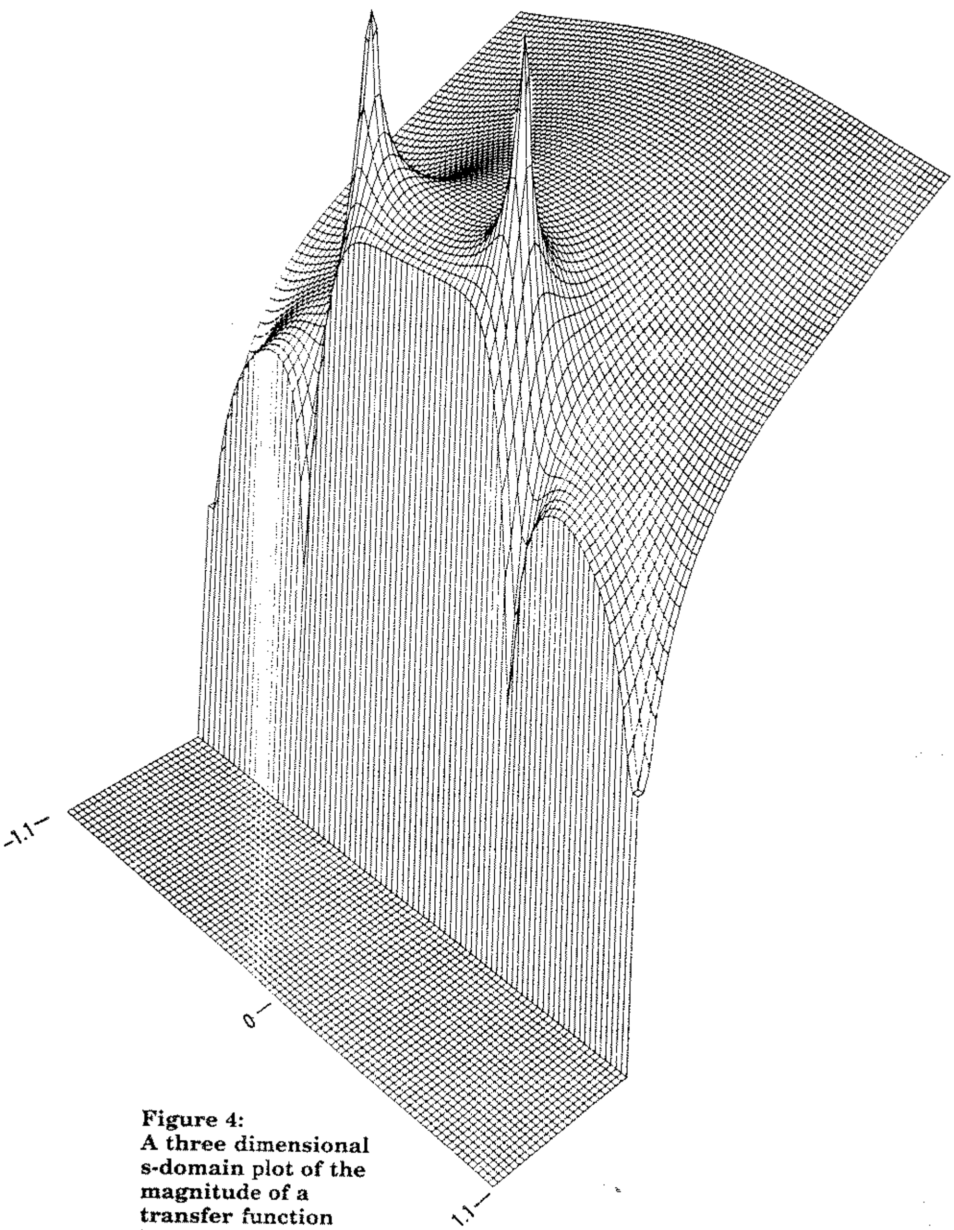


Figure 4:
A three dimensional s-domain plot of the magnitude of a transfer function having two poles and four zeros, with the right half of the plane cut away to show the frequency response function along the imaginary axis.

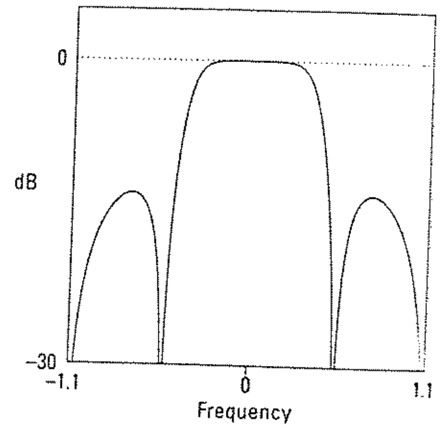


Figure 5:
A plot of the frequency response function illustrated in figure 4.

As a brief review and comparison, figure 2 shows the conventional s-domain, and figure 3 shows the z-domain with the unit circle drawn. Figure 4 shows a three-dimensional view of an analog filter in the s-domain, with the right half of the plane removed, showing the frequency response function along the imaginary axis. Figure 5 is a plot of this frequency response function. This filter comprises a pair of poles in the left half plane, along with 4 zeros along the frequency axis.

Figure 6 shows a three-dimensional z-domain view of a digital filter with characteristics similar to those of the analog filter. Here, the outside of the unit circle has been removed to show the frequency response function around this circle. Figure 7 shows a plot of this frequency response function. This filter comprises a pair of poles inside of the unit circle forming the filter pass-band, and three zeros on the unit circle forming the stop-band.

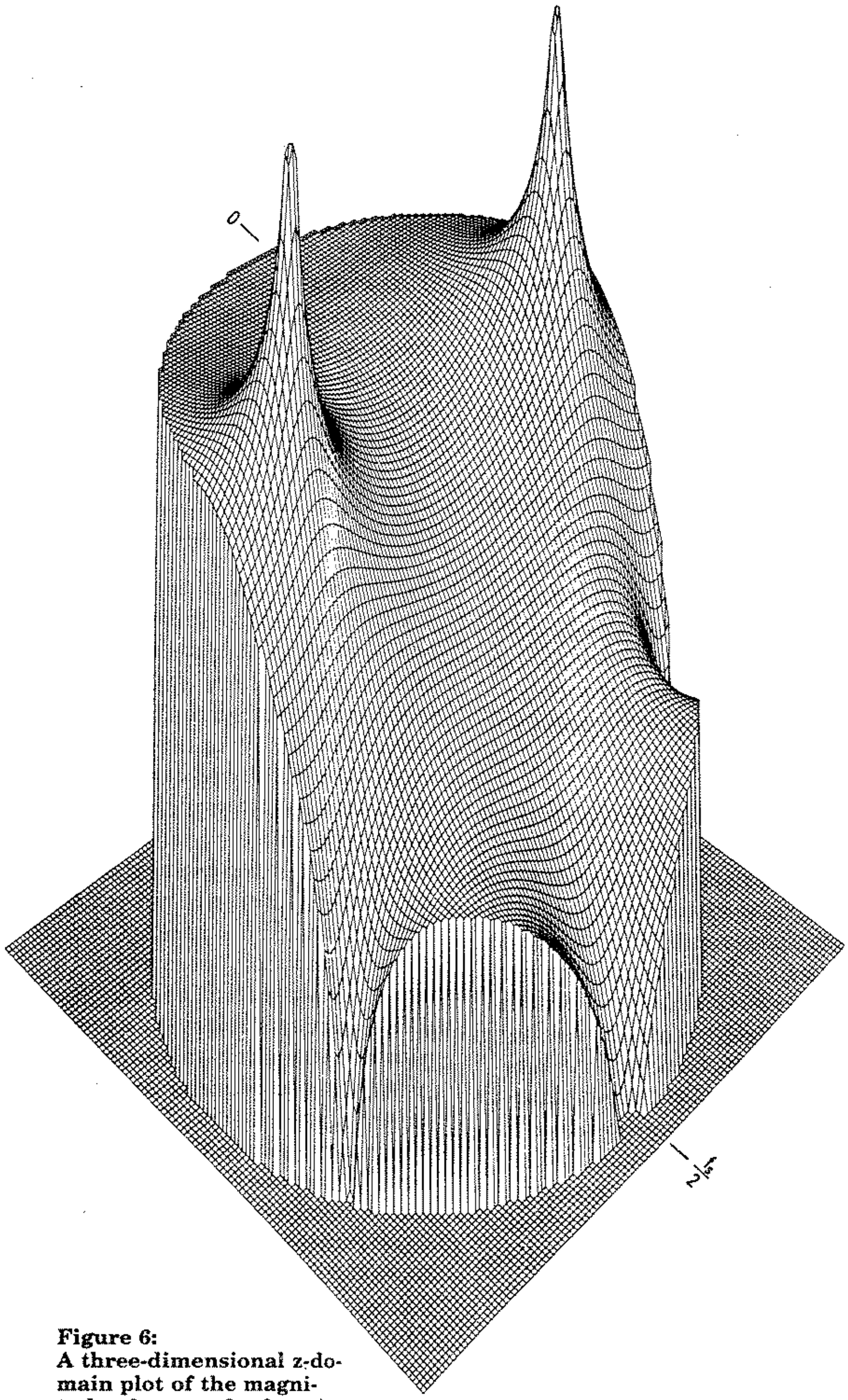


Figure 6:
A three-dimensional z-domain plot of the magnitude of a transfer function having two poles and three zeros, with the exterior of the unit circle cut away to show the frequency response characteristic.

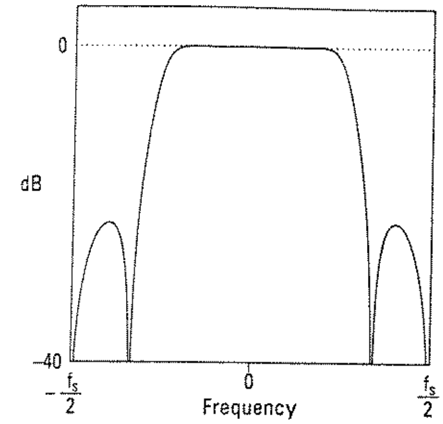


Figure 7:
A plot of the frequency response function illustrated in figure 6.

The z-domain is particularly useful for representing the behavior of digital filters, which involve combinations of adders, multipliers, and sample delay registers, since 1/z represents one sample of time delay. The equation describing the performance of a digital filter comprises the ratio of two polynomials in z, just as the performance of an analog filter comprises the ratio of two polynomials in s.

In the next section, the z-transform is derived from the Laplace transform and various techniques for converting from one domain to the other are discussed. In the final section, the differences between the two domains will be discussed, along with some of the problems that are encountered in mixed domain measurements.

Chapter 1: Derivation of the z-Transform from the Laplace Transform

The Laplace transform H(s) of some system whose impulse response is h(t) is given by

$$H(s) = \int_0^\infty h(t) e^{-st} dt \qquad (1)$$

where s is the Laplace variable. The impulse response is assumed to be zero for negative time values.

In a sampled data system with a sampling rate of f_s, the sample interval in the time domain is $\Delta t = 1/f_s$. A sampled version of h(t) can be obtained by multiplication by the "Shah" function (see reference [1]) defined by

$$III(t/\Delta t) = \Delta t \sum_{k=-\infty}^{\infty} \delta(t - k\Delta t) \qquad (2)$$

where $\delta(t-k\Delta t)$ is the unit impulse or delta function centered at $t = k\Delta t$. The area under this delta function is unity. If this sampled version of h(t) is inserted into (1), and the orders of integration and summation are interchanged, the resulting s-domain transfer function for this sampled system becomes

$$H(s) = \Delta t \sum_{k=0}^{\infty} h(k\Delta t) e^{-ks\Delta t} \qquad (3)$$

Make the substitution

$$z = e^{s\Delta t} \qquad (4)$$

Then the z-transform of the system impulse response is

$$H_z(z) = \frac{H(s)}{\Delta t} = \sum_{k=0}^{\infty} h(k\Delta t) z^{-k} \qquad (5)$$

The quantity z^{-k} is the Laplace transform of a delta function delayed by $k\Delta t$ in the time domain. The coefficient on the kth power of 1/z is simply the kth sample of the impulse response. Note that the sampling interval Δt has been removed as an amplitude multiplying factor from the definition of the z-transform in (5). This factor must be restored to evaluate the frequency response along the unit circle.

The periodic nature of the transform of sampled time data along the frequency axis can be seen from (3), where an exponential in continuous time has been replaced by an exponential involving multiples of the sampling interval Δt. Whenever s is replaced by $s+i2\pi n/\Delta t$, for any integer n, the value of the transform is unchanged. Figure 8 shows the frequency response for a simple pole at $s = -0.1$ (solid curve), and the four images obtained by evaluating the z-transform of the impulse response around the unit circle (dashed curve).

Equation (4) completely defines the z-domain in terms of the s-domain, and it is apparent that there is no new information about the transfer function contained in the z-domain representation. In fact, the z-domain form actually contains less information than the original s-domain form, to the extent that the original frequency response bandwidth exceeds half of the sampling rate. Any higher frequency components have been replaced by periodic replication of the lowest order image or, from another perspective, continuous time data has been replaced by sampled data. The loss of information is also apparent from equation (4), where a value for z is always uniquely determined for any given value of s, but the converse is not true. Thus, the merit of the z-domain is that it only shows the available information about the transfer function, whereas the s-domain may show redundant information.

Unfortunately, if equation (4) is used to obtain the z-transform directly from the Laplace transform, a rational fraction in s (comprising poles and zeros in the s-domain) becomes a transcendental function in z. For example, a simple pole in the s-domain can be written in the z-domain as

$$\frac{1}{s+a} = \frac{\Delta t}{ln(z) + a\Delta t} \qquad (6)$$

Any hardware implementation of a z-domain digital filter comprises various combinations of adders, multipliers, and sample delays represented by integer powers of 1/z. Thus, the z-domain form of the system transfer function must comprise finite order polynomials in z, and hence can be represented either as a rational fraction or a partial fraction in z. The transcendental form shown in (6) cannot be easily implemented physically. This argument implies that any practical transformation between the s and z domains must be only approximate. This raises the question as to the amount of error introduced by the approximation. Some of the more common transformations between these two domains will be discussed next, and some examples of the associated errors will be given.

Chapter 2: Transformations Between s and z Domains

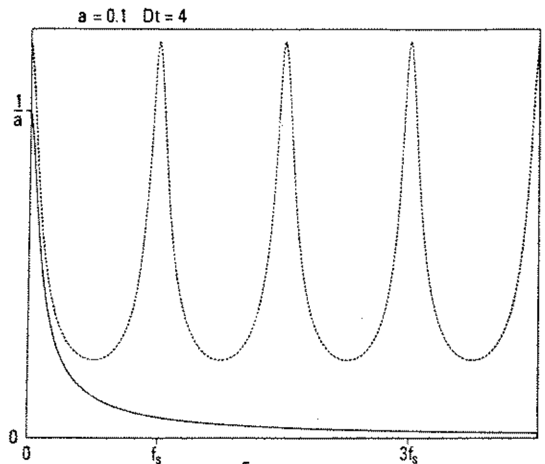


Figure 8: The frequency response function of a simple pole in the s-domain (solid line), compared with the frequency response of the impulse invariant form of the z-transform (dashed line). Note the four images introduced by the sampling operation. Also note the error in peak amplitude.

There are two generic types of transforms between the s and z domains, with numerous variations on each method. The first type involves matching time waveforms, usually either the system impulse response or the step response. The second type involves rational fraction approximations to equation (4), such as given by the bilinear transformation.

None of the above methods are exact, and the choice between them depends upon the application at hand. The common link between analog and digital parts of a mixed system is either the frequency response function or the impulse response in the time domain. Thus, the significance of any errors introduced by approximations between the s and z domains will ultimately be viewed along either the frequency or the time axis. Because of aliasing in sampled systems, it is often not possible to match both the frequency response function and the impulse response simultaneously. In general, either the impulse invariant or the step invariant methods are best when the time response is of interest. The bilinear transform is used for frequency response matching, but is only accurate for very low frequencies, relative to the sampling frequency f_s.

2.1: Impulse Invariant Transformation

The impulse response can be matched by decomposing the transfer function in either the s or the z domain into partial fractions, then matching the impulse response of each term. This is easy to do if the multiplicities of all poles are unity, but becomes more complicated for multiple poles. The multiple pole case is discussed in the Appendix. A simple pole is represented in the s-domain by

$$H(s) = \frac{1}{s+a} \qquad (7)$$

and the corresponding impulse response is

$$h(t) = e^{-at}, \; for \; t > 0 \qquad (8)$$

From (5), the z-transform can be written as

$$H_z(z) = \sum_{k=0}^{\infty} e^{-ak\Delta t} z^{-k} \qquad (9)$$

$$= \frac{z}{z - e^{-a\Delta t}} \qquad (10)$$

Thus, each partial fraction term in the s-domain with a pole at s = –a yields a partial fraction term in the z-domain, with a zero at the origin and a pole at z = exp(–aΔt). The sampled values of the impulse response become the coefficients on an infinite series in 1/z, as shown in equation (9), which can be written in closed form (10).

The frequency response corresponding to the z-domain transfer function is obtained by multiplying H_z by Δt, for z=exp(i2πfΔt). If Δt is sufficiently small, then this becomes

$$H(i2\pi f) = \frac{1}{a + i2\pi f} \qquad (11)$$

which is the same as obtained from the s-domain via (7). However, when Δt is not sufficiently small, the z-domain frequency response is different from the s-domain response. This is a direct result of the aliasing that occurs in the frequency domain when images of the frequency response function are replicated at multiples of the sampling frequency.

2.2: Step Invariant Transformation

In a similar manner, it is possible to match the response to a unit step. The s-domain transfer function is multiplied by 1/s, and the result is expressed in partial fraction form. Then, each term is converted to the z-domain, as indicated above, and multiplied by (z-1)/z to remove the input step.

If this technique is used to match the step response for a simple pole, as given by (7), the result is

$$H_z(z) = \frac{A}{z - e^{-a\Delta t}} \qquad (12)$$

where

$$A = \frac{1 - e^{-a\Delta t}}{a\Delta t} \qquad (13)$$

Compared to (10), this transfer function has only a pole at z = exp(–aΔt), and no finite zeros.

None of these techniques that match responses in the time domain consider the effects of aliasing caused by undersampling. Thus, even though the time response is matched at the sample values, any waveform details that may occur between samples, such as fast level transitions or narrow pulses, are lost. This implies that the higher frequencies in the frequency response function may be in error to some degree. This is a direct consequence of the potential overlap between the replicated frequency images that result from time domain sampling.

Only partial fraction terms that involve poles or a constant can be precisely converted from one domain to another. Thus, any higher order polynomial components that result from the partial fraction expansion cannot be converted. This means that the order of the numerator of the rational fraction form must be no greater than the order of the denominator for either an impulse invariant or a step invariant conversion to exist. An exception to this rule can be made for any powers of z that can be removed from the rational fraction before conversion, since these powers of z can be represented as time advances. This is also true for powers of 1/z which can be represented as time delays.

2.3: Bilinear Transformation

The second type of transformation between the s and z domains is used when frequency response function matching is needed at low frequencies, and involves some sort of rational fraction approximation to equation (4). The most common approximation is called the bilinear transform, which is obtained from the quotient of two first order polynomials in s. Equation (4) can be written as

$$z = e^{s\Delta t} = \frac{e^{s\Delta t/2}}{e^{-s\Delta t/2}} \qquad (14)$$

If only the first two terms in the Taylor's series expansion of the numerator and the denominator are retained, then z can be approximated by

$$z \cong \frac{1 + s\Delta t/2}{1 - s\Delta t/2} \qquad (15)$$

This can be inverted to obtain

$$s \cong 2/\Delta t \frac{z-1}{z+1} \qquad (16)$$

Equation (16) is called the bilinear transform (the quotient of two linear expressions), and (15) is sometimes called the inverse bilinear transform. This form has the advantage of limiting the orders of the z-domain polynomials to the maximum order of the s-domain polynomials. Obviously, there are many other possible polynomial approximations to (4), but this is the one most often used in practice.

The bilinear form also has the property of mapping the entire s-domain frequency axis onto the unit circle in the z-domain, in contrast to the exact definition of z, in which only frequencies up to half of the sampling rate are mapped onto the unit circle. Unfortunately, this mapping results in a considerable amount of frequency "warping", especially for frequencies near the point z = –1. This warping is described by

$$f' = \frac{\tan(\pi f \Delta t)}{\pi \Delta t} \qquad (17)$$

where f' is the frequency after the bilinear transform has been imposed, and f is the frequency around the unit circle in the z-domain at which f' is mapped. Note that f' becomes infinite when f = 1/(2Δt) = half of the sampling frequency.

For purposes of comparison, the expression for s in (16) can be substituted into (7) to obtain the bilinear form for a simple s-domain pole. The result is

$$H_z(z) = \frac{H(s)}{\Delta t} = B\frac{z+1}{z-b} \qquad (18)$$

where

$$B = \frac{1}{2 + a\Delta t} \qquad (19)$$

$$b = \frac{2 - a\Delta t}{2 + a\Delta t} \qquad (20)$$

Thus, a pole is placed at z = b and a zero is placed at z = –1. If the sampling interval Δt is sufficiently small, z can be replaced by 1+i2πfΔt and b is approximately 1–aΔt, so H(i2πf) becomes 1/(a+i2πf), as expected. However, a comparison of (18) with (10) shows that these equations are not equivalent and, therefore, the frequency response and the impulse response will be different.

These transformation techniques that involve approximations to (4) tend to include the effects of aliasing to some extent, but the resulting responses in the time domain may not be very accurate. These approximations are only good for small values of s, for which z is near unity.

2.4: Representation of Time Delays

Any time delay in the s-domain representation of a system must be carried as a separate parameter since there is no finite rational fraction representation of this delay. However, in the z-domain, integer multiples of the sampling interval Δt are represented as powers of 1/z, which are simply poles at the origin in the z-plane. Thus, these discrete time delay values can be represented as part of a z polynomial. Unfortunately, this technique does not work for time delays that are fractions of the sampling interval, so it is still necessary to carry a time delay parameter separately. One possible convention is to always represent the integer time delay multiples of Δt as z-domain poles at the origin, and to represent only the fractional part of the delay as a separate parameter. However, this is an arbitrary choice, and other conventions for representing delay are equally valid.

In any case, when a z-domain transfer function is converted into an s-domain representation, the resulting time delay is the sum of the part represented by a multiple pole at the origin of the z-domain, and the part represented as a separate delay parameter.

2.5: Comparison of Different Transformation Techniques

The results of each transformation technique can be compared by viewing the amplitude and phase response in the frequency domain, and/or the impulse response in the time domain. Figure 9 shows the amplitude frequency response of a simple pole in the s-domain for a = 0.1 (solid curve), along with curves evaluated from three z-domain approximations (dashed curves). The upper dashed curve is for the impulse invariant transformation, and the middle dashed curve is for the step invariant case. The lower dashed curve is for the bilinear transform. Notice the zero at z = −1 for the bilinear case. Also note that the dc value of the response is not correct for the impulse invariant case, although when this is normalized away, this curve coincides with the step invariant curve.

Figure 10 shows the phase response for the same four cases illustrated in figure 9. The solid line represents the phase for the s-domain representation of a simple pole, while the dashed lines represent the phase for three different z-domain approximations. It is only necessary to consider the phase angle for positive frequencies below half of the sampling rate (left half of the figure) since the negative frequency interval will be symmetric. The best phase match to the solid line is obtained by means of the bilinear transform (middle dashed line). This is expected since the bilinear transform incorporates, to some extent, the effects of aliasing. The upper dashed line is for the impulse invariant case, and the lower dashed line is for the step invariant case. The step invariant case incorporates an extra phase slope that corresponds to one sample of delay. If this delay is removed, this case is identical to the impulse invariant case.

Figure 11 shows the impulse responses in the time domain for these same four cases. The solid line is the continuous time impulse response for a simple pole, and the labels *, o, and x show the sampled versions of this impulse response for the impulse invariant method (*), the step invariant method (o), and the bilinear transformation (x). The impulse invariant method gives exact sample values. If the step invariant results were re-scaled in amplitude, they would also be correct except for one sample of delay. The bilinear transform results need to be scaled in amplitude, and the decay time constant is also slightly in error (too small by 1.348%, for this case).

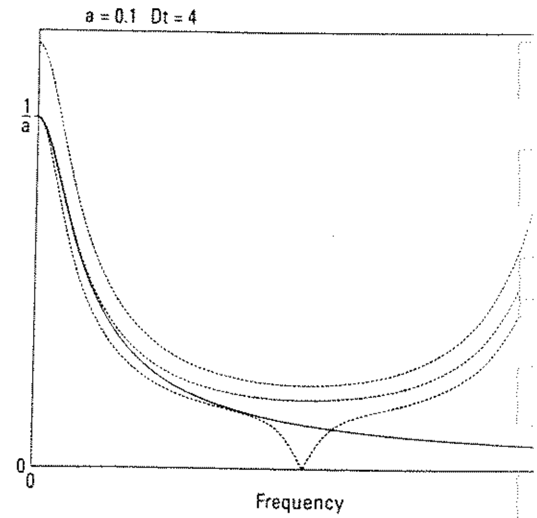


Figure 9:
The frequency response function of a simple pole in the s-domain (solid line), compared with the frequency responses for three different z-domain representations. The upper dashed line is for the impulse invariant case, and the middle dashed line is for the step invariant case. The bilinear transform case is shown by the lower dashed line. When the upper line is scaled to be correct at dc, it matches the middle line at other points, as well.

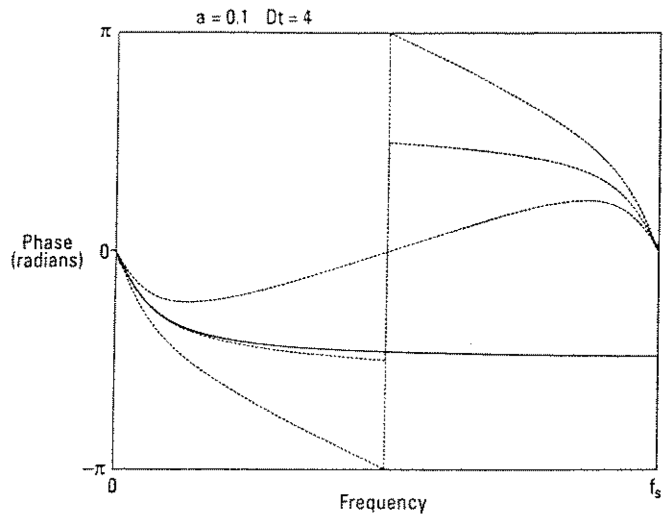


Figure 10:
The phase response of a simple pole in the s-domain (solid line), compared with the phase responses of three different z-domain representations. Only the left half of the plot is useful for this comparison. The upper dashed line is for the impulse invariant case, and the lower dashed line is for the step invariant case. The middle line is for the bilinear transformation.

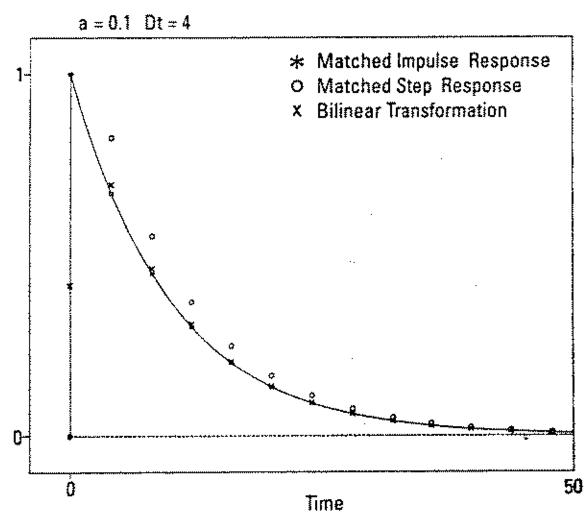


Figure 11:
The impulse response of a simple pole in continuous time (solid line), and for sampled times corresponding to the impulse invariant method (* symbol), the step invariant method (o symbol), and the bilinear transform technique (x symbol).

Chapter 3: Characteristics of Mixed Domain Measurements

When continuous time and sampled time systems are connected together as shown in figure 1, there arises the need to make frequency response measurements across the interface between the two domains. When making measurements in a mixed analog/digital system, the key characteristics to be aware of are:

- The occurrence of multiple spectral images of sampled signals
- The need to synchronize analog and digital sampling rates
- The possible presence of two types of time delays

In mixed domain systems, multiple images occur in the spectra of sampled signals. It is generally necessary to filter the input signals to reduce aliasing, and to filter the output signals to attenuate the spectral images. The bandwidth of the measurements of analog signals must extend beyond the frequency of the highest image of concern.

There is also a need to synchronize the analog and digital sampling rates to avoid errors due to leakage. In addition, these sampling signals must be phase locked so that transfer functions between digital and analog parts of a system can be measured accurately. If there is any relative jitter between these two sampling signals, then additional errors will be introduced.

Two types of time delay appear in a mixed mode measurement, in contrast to only one type of delay in an analog measurement. In either case there can be a delay in the system impulse response, but there can be an additional delay in the sampling pulses for the digital part of a system. These two delays affect the results in different ways.

3.1: Images and Analog Filtering

The distinguishing feature of mixed domain measurements is the occurrence of multiple images in the spectrum of a sampled signal. Generally, a designer is interested in the effect that an analog filter circuit has upon the multiple images introduced by the digital portion of a system. An ADC is an example in which aliasing is introduced into the primary spectral image if any input signal components occur at frequencies above half of the sampling rate. Attenuating such unwanted signals is the purpose of the low-pass anti-aliasing filter in figure 1. To observe higher frequency signal components, the bandwidth of a measurement on the analog input to an ADC must extend beyond the frequency encompassed by the highest image of concern in the sampled signal.

In a similar manner, the analog filter on the output of the DAC in figure 1 is designed to attenuate the higher order images coming out of the mixed system. This filter also serves to convert the discrete samples of the DAC output into a continuous analog signal. To show all of the attenuated images of interest, it is necessary to make analog measurements at frequencies higher than the digital sampling rate.

This latter filter must be designed to attenuate all of the frequency domain images of the spectrum except the one of interest. One common filter type is that obtained by means of a zero order hold circuit. The impulse response of this filter is a rectangle having a unit area and a width equal to the sample interval Δt. This gives a filter shape of $\sin(\pi f \Delta t)/(\pi f \Delta t)$, which has nulls at the center of each image except the one centered at the origin. This filter shape is shown as a dashed line in figure 12 and its effect upon the frequency images (of figure 8) is shown as a solid line. In addition, there will be a linear phase shift versus frequency corresponding to a delay of $\Delta t/2$.

Other filters must generally be added to further reduce the sizes of the unwanted images. It is apparent from figure 12 that these reduced images can still be relatively large. It is possible to use higher order hold circuits, corresponding to triangular or parabolic impulse responses, but it is usually easier to design one of the standard analog low-pass filters such as either the Chebyshev or elliptic types.

Often, mixed mode systems are designed so that the digital filter part complements the analog part to obtain better overall characteristics than could be obtained with either technique separately. For example, the digital filter might be designed with a narrow pass-band relative to the sampling frequency, or the sampling frequency might be multiplied, so that the subsequent analog filter can have a wider transition band between the desired image and the remaining rejected images. This oversampling allows use of a simple analog filter design, having well controlled phase and amplitude characteristics.

3.2: Synchronization of Sampling Pulses

The second key characteristic of mixed domain measurements is the need for synchronized sampling pulses between the two domains. Not only should the sample rates be related by simple integers, but the relative phases between the two sampling signals must be known or measured so that mixed domain transfer functions can be determined.

Generally, the analog sampling rate will be some integer multiple of the digital rate. However, the digital rate is often determined by the device under test, so the analog sampling signal must be derived from the digital rate in some manner. To minimize leakage effects when working with periodic signals, the analog sampling rate must be a very accurate multiple of the digital rate. Thus, the digital rate must be known or measured very accurately, and the analog sampling rate must be very accurate and stable in frequency.

There are two types of errors that can occur when timing differences exist between the analog and digital sampling signals. The first type of error is due to a discrepancy in the average analog sampling rate (not exactly an integer multiple of the digital rate). In this situation, a periodic digital signal will not remain exactly periodic after being sampled at the analog rate. When using a control systems analyzer or dynamic signal analyzer to make measurements in this situation, the sampling delays can cause leakage errors in the frequency spectrum, especially if the user selects a rectangular or "uniform" window. For example, when making distortion measurements using a sinusoidal input, a uniform window is generally used and all harmonics are expected to be exactly periodic in the time window. Any leakage that occurs will directly affect the accuracy of measurements of the higher order harmonics.

The second type of error is due to jitter on the digital sampling signal. There are times when this same jitter should also occur on the analog sampling signal. For example, if the transfer function of a DAC is being measured, then any jitter on the digital samples should be exactly duplicated on the analog samples so that the measured transfer function is independent of this jitter. However, if a digital compensator is embedded in a control system and there is some amount of jitter on the internal digital clock, then the analog sampling rate should probably be uniform in time so that the effects of the digital jitter can be observed.

Jitter on the digital sampling signal can also result in leakage errors, especially if a uniform time window is used in the measurement. For example, if the transfer function of a DAC with zero order hold is being measured using a uniform window and there is jitter on the digital clock, then leakage contributions from the higher order images will occur in the baseband frequency region, even if the analog sampling rate is much higher than the digital rate (negligible aliasing).

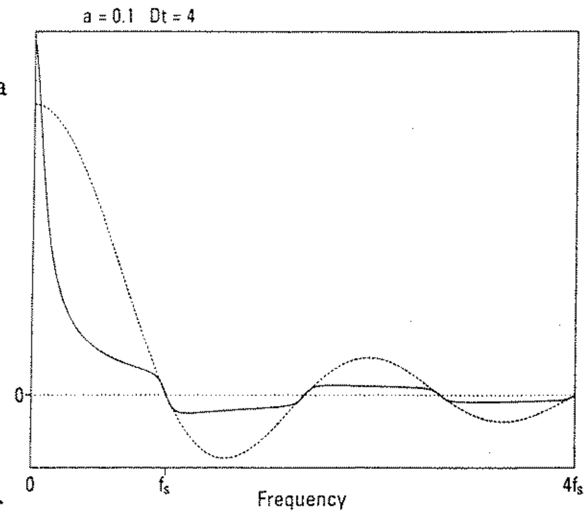


Figure 12: The equivalent filter of a zero order hold is shown as a dashed line, and the effect of this filter on the multiple frequency images of figure 8 is shown as a solid line. Note the nulls in this filter at the center of each image.

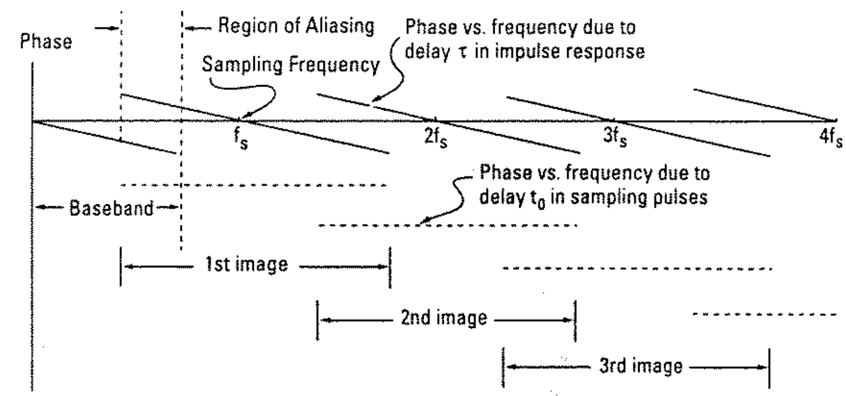


Figure 13: Phase versus frequency due to a time delay in the impulse response, and due to a time delay in the sampling pulses.

Chapter 4: Summary

3.3: Time Delays

The third major characteristic of mixed measurements that must be considered is the occurrence of time delays in the system. In the analog part of a system, a time delay results in a linear phase slope in the frequency response function and can be approximated by a rational fraction in the s-domain. In the z-domain, there are two types of time delays that must be treated separately. There can be time delays in the signal path, just as for analog systems, and there can be time delays in the sampling pulses, without any signal delay. In addition, both kinds of delay may occur simultaneously. Delays in the sampling pulses can occur if multi-phase clocks are used to perform several operations within one clock period, particularly if the output is clocked with a different phase than the input.

If there is a delay in the signal path, then the result is the same as for an ordinary analog delay. A linear phase slope is introduced into the frequency response function (see figure 13). A modified z-transform can be defined (see reference [2]) that matches the delayed impulse response, although the linear phase slope in the frequency response may not be correctly represented due to aliasing. Alternatively, a rational fraction in z can be used to approximate the phase slope, just as in the s-domain.

A delay in the sampling pulses only affects the phases of the higher order images of the frequency spectrum, and hence only affects the errors due to aliasing. If the original spectrum is band limited to half of the sampling frequency, then a delay in the sampling pulses has no effect upon the baseband spectral image (see figure 13). In the z-domain, the coefficients on the powers of 1/z are obtained from delayed samples of the impulse response, so the actual z-transform is modified by the sample delay. In addition, there is a factor of z^{-d}, where d is the sample time delay normalized by the sample interval Δt, which accounts for the phase differences among the frequency images.

When both the sampling pulses and the impulse response are delayed, the result is a combination of the effects discussed above for each separate delay. However, if both signals are delayed by the same amount, then the samples of the impulse response are the same as for no delay, and the resulting z-transform only differs by the z^{-d} factor defined above.

These time delay effects are best summarized by re-writing equation (5) for the z-transform, where h(t) has been replaced by $h(t-\tau)$, to represent a delay of τ in the impulse response, and t has been replaced by $t-t_0$ in the Shah function (equation (2)) to indicate a sampling pulse delay of t_0. The resulting z-transform can be expressed as

$$H_z(z) = \sum_{k=0}^{\infty} h(k\Delta t + t_0 - \tau) z^{-k-d}, \quad (21)$$

$$d = \frac{t_0}{\Delta t}$$

Notice that these two types of delay enter into the equation in different ways, so their effects must be considered separately.

There are numerous applications involving mixed analog and digital signals in the same system. It is helpful to use the z-transform for the digital part, in conjunction with the Laplace transform (s-domain) for the analog part when making measurements on these mixed systems. The z-transform is defined and the impulse invariant, step invariant and bilinear transformations between the s and z domains are discussed and compared. The Appendix discusses the matching of the impulse responses of multiple poles in both the s and z domains.

Three key characteristics of mixed domain measurements are discussed: images and analog filtering; synchronized sampling; and time delay effects. Most mixed analog/digital systems contain analog filters on the input of ADCs to prevent aliasing and on the output of DACs to attenuate images. When using a control systems analyzer or dynamic signal analyzer to measure the higher order images in a mixed transfer function, the analog sampling rate should be some integer multiple of the digital rate. To make accurate frequency response measurements, these two sampling rates must be carefully locked together in both frequency and phase. There is also the need to handle time delays, both in the signal path and in the sampling pulse path.

Appendix: Multiple Pole Impulse Invariance

The transfer function of a multiple pole in the s-domain is

$$H(s) = \frac{1}{(s+a)^{k+1}} \qquad \text{(A1)}$$

The pole is located at s = -a, and it has a multiplicity of k+1. The corresponding impulse response is given by

$$h(t) = \frac{t^k}{k!} e^{-at}, \quad for \ t \geq 0,$$
$$k = 0, 1, 2, \ldots \qquad \text{(A2)}$$

The goal is to derive a z-domain representation that will exactly reproduce this impulse response at times sampled at Δt intervals. In particular, the sampled impulse response is given by

$$h(n\Delta t) = \frac{(n\Delta t)^k}{k!} e^{-an\Delta t},$$
$$for \ n = 0, 1, 2, \ldots \qquad \text{(A3)}$$

This sampled impulse response can be generated from a z-domain formulation involving the sum of poles having all multiplicities from unity to k+1. The detailed derivation will not be given here, but the results for poles of multiplicity one through four will be shown. In general, whenever a pole of a given multiplicity occurs, all poles of lower order also occur. Thus, a matrix representation of this impulse invariant transformation is useful.

Define a normalized z-domain variable called x, as follows

$$x = \frac{e^{-a\Delta t}}{z} \qquad \text{(A4)}$$

Define a four element vector S whose elements are the s-domain poles for each multiplicity. Define Z as 4-vector of z-domain poles of the form $1/(1-x)^{k+1}$ for each multiplicity. The elements for each of these vectors are listed in order of decreasing multiplicity. Then, the impulse invariant transformation between these two domains can be written in matrix form as

$$S \Leftrightarrow RZ \qquad \text{(A5)}$$

where R is the 4x4 matrix

$$R = \begin{bmatrix} \Delta t^3 & 0 & 0 & 0 \\ 0 & \Delta t^2 & 0 & 0 \\ 0 & 0 & \Delta t & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & -2 & 7/6 & -1/6 \\ 0 & 1 & -3/2 & 1/2 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad \text{(A6)}$$

If a row vector A of s-domain coefficients on each element of S is defined as

$$A = [A_3 \quad A_2 \quad A_1 \quad A_0] \qquad \text{(A7)}$$

then the final result can be written as

$$AS \Leftrightarrow ARZ \qquad \text{(A8)}$$

In a similar manner, equation (A5) can be inverted to give

$$Z \Leftrightarrow R^{-1}S \qquad \text{(A9)}$$

where the inverse of R is

$$R^{-1} = \begin{bmatrix} 1 & 2 & 11/6 & 1 \\ 0 & 1 & 3/2 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \Delta t^{-3} & 0 & 0 & 0 \\ 0 & \Delta t^{-2} & 0 & 0 \\ 0 & 0 & \Delta t^{-1} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad \text{(A10)}$$

If a row vector B of z-domain coefficients on each element of Z is defined as

$$B = [B_3 \quad B_2 \quad B_1 \quad B_0] \qquad \text{(A11)}$$

then the result can be written as

$$BZ \Leftrightarrow BR^{-1}S \qquad \text{(A12)}$$

These row vectors of coefficients are related by

$$B = AR \qquad \text{(A13)}$$

If the multiplicity of the original pole is reduced by one, then the topmost row and the leftmost column of R (and of R^{-1}) are discarded to form a 3x3 R (and R^{-1}) matrix.

As for the unity multiplicity case, it is necessary to multiply the z-domain form of the multiple pole by Δt before attempting to calculate the frequency response function around the unit circle.

References

[1] Bracewell, R.N., The Fourier Transform and Its Applications, McGraw-Hill, 1986.

[2] Phillips, C.L., and Nagle, H.T., Digital Control System Analysis and Design, Prentice-Hall, 1984.

HEWLETT PACKARD

For more information, call your local HP sales office listed in your telephone directory or an HP regional office listed below for the location of your nearest sales office.

United States:
Hewlett-Packard Company
4 Choke Cherry Road
Rockville, MD 20850
301 670 4300

Hewlett-Packard Company
5201 Tollview Drive
Rolling Meadows, IL 60008
312 255 9800

Hewlett-Packard Company
5161 Lankershim Blvd.
No. Hollywood, CA 91601
818 505 5600

Hewlett-Packard Company
2015 South Park Place
Atlanta, GA 30339
404 955 1500

Canada:
Hewlett-Packard Ltd.
6877 Goreway Drive
Mississauga, Ontario L4V1M8
416 678 9430

Australia/New Zealand:
Hewlett-Packard Australia Ltd.
31-41 Joseph Street
Blackburn, Victoria 3130
Melbourne, Australia
03 895 2895

Europe/Africa/Middle East:
Hewlett-Packard S.A.
Central Mailing Department
P.O. Box 529
1180 AM Amstelveen
The Netherlands
31 20/547 9999

Far East:
Yokogawa-Hewlett-Packard Ltd.
29-21, Takaido-Higashi 3-chome
Suginami-ku, Tokyo 168
03 331-6111

Latin America:
Latin American Region Headquarters
Monte Pelvoux Nbr. 111
Lomas de Chapultepec
11000 Mexico, D.F. Mexico
905 596 79 33

Data subject to change.

Printed in USA 08/89
5952-7250

z-Domain Curve Fitting in the HP 3563A Analyzer

HP 3563A-1 Product Note

Introduction

This product note is a supplement to HP product note 3562A-3, entitled "Curve Fitting in the HP 3562A." That previous note described the s-domain curve fitter, used in both the HP 3562A and the HP 3563A. However, the HP 3563A has the additional capability of accepting digital input signals, and hence must be able to represent transfer functions of digital systems in the z-domain. HP application note 243-4 entitled "Fundamentals of the z-Domain and Mixed Analog/ Digital Measurements," describes the z-domain and discusses the nature of mixed measurements involving both s and z domain representations. One of the new capabilities in the HP 3563A is a z-domain curve fitter that can be used to determine a z-domain representation of a system from a measured frequency response function.

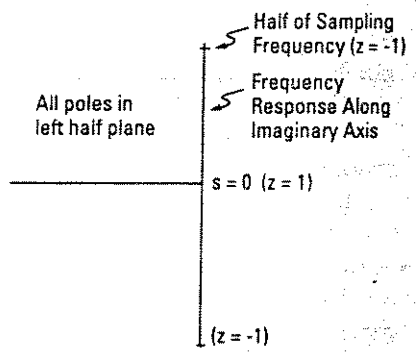


Figure 1:
The s-plane, showing the imaginary frequency axis and half of the sampling frequency for a digital system. All poles must be in the left half of this plane for a stable system.

Figure 1 shows a representation of the s-plane used to display poles and zeros of transfer functions of analog filters. All poles must lie in the left half of this plane for a stable system. The frequency response is represented along the imaginary axis. Figure 2 shows the z-plane, with the frequency response represented around the unit circle. In this domain, all poles must lie inside of the unit circle for system stability. The z variable is related to s by $z=\exp(s\Delta t)$, where Δt is the time interval between sample points in a digital measurement. The z-plane is used to display poles and zeros of transfer functions of digital filters.

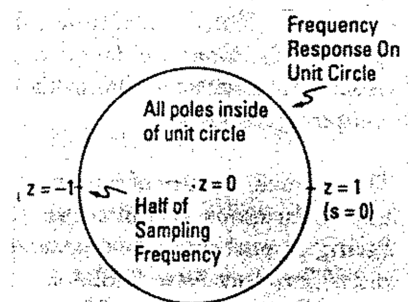


Figure 2:
The z-plane, showing the frequency axis around the unit circle, with half of the sampling frequency at z=-1. All poles must lie inside of the unit circle for a stable system.

The basic theory of curve fitting in the s-plane is described in reference [1].

In general, curve fitting comprises the calculation of the coefficients needed to multiply a set of basis functions so that the sum of these weighted basis functions fits the measured data in some optimum way. These basis functions are selected to represent some mathematical model of the physical system in question and are often selected to be mutually orthogonal, which means that the coefficient for each function of the set can be calculated independently of all of the other coefficients. In reference [1], the basis functions are Chebyshev polynomials as functions of frequency. The theory behind z-domain curve fitting is very similar except that the basis functions are complex exponentials as functions of frequency, corresponding to powers of z around the unit circle. These basis functions are periodic along the frequency axis, with a period equal to the sampling frequency.

The general nature of estimation techniques, and of curve fitting in particular, is discussed in HP Product Note 3562A-3, and will not be repeated here. However, the points made in that note are equally applicable to the z-domain curve fitting process, so it will be assumed that this introductory material has been consulted.

Some of the key points to remember are:

- A curve fitter is used to estimate the coefficients of a rational fraction representation (in either the s or z domain) of a measured frequency response function. The resulting poles, zeros, and gain factor can be obtained from these coefficients.
- Curve fitting will be very difficult and inaccurate unless a rational fraction model of the system is essentially valid.
- The coefficients (and hence the poles, zeros, and gain factor) are random quantities to some extent, depending upon the amount of noise or uncertainty in the measured data.
- A weighting function is applied to the measured data, to emphasize regions near peaks and valleys, and to de-emphasize regions having poor signal-to-noise ratio, or regions having excessive distortion, aliasing or interference.
- Curve fitting is still somewhat of an art, so a strict "cookbook" approach is not very practical. There are numerous factors that can cause the fitter some degree of difficulty and the user should be aware of these.

Just as an s-domain curve fitter is used to obtain an analytical model of some measured transfer characteristic in the s-domain, a z-domain curve fitter is used to obtain an analytical model of some measured transfer characteristic in the z-domain. Ideally, to obtain the best results from a z-domain fitter, the measured characteristic should be the frequency response of a digital filter. However, there are cases when a digital representation of an analog filter is desired. For example, a digital filter might be required to replace an existing analog compensator in a closed-loop control system. Fitting a z-domain model to an analog frequency response is more difficult than fitting to a digital response, but it can be done if extra poles and zeros are allowed.

It is possible to use the s-domain fitter on a measured frequency response to obtain an s-domain model, and then to use one of the standard transformations to convert this model into the z-domain. The HP 3563A analyzer supports three conversion transformations: the impulse invariant, the step invariant, and the bilinear transformation. Unfortunately, none of these transformations are completely accurate, so some errors will result.

This conversion from the s-domain to the z-domain can often be done better by fitting the measured data directly, using the z-domain curve fitter. This approach will generally give good fit quality over the entire frequency range of interest. In addition, the measured data can be manipulated (via MATH operations) to remove errors or to compensate for other components in the system, such as time delays or zero order hold characteristics, before the fit is calculated. Also, the weighting function in the fitter can be used to emphasize frequency regions of interest, and to ignore regions that are not important. It might even be possible to obtain a good fit with reduced polynomial orders. The main disadvantages to this method are the potential for poles outside of the unit circle, and the possibility of obtaining a non-minimum phase transfer function. There is also no explicit control over the resulting filter time response.

The remainder of this note will be devoted primarily to a description of how to use the z-domain curve fitter in the HP 3563A analyzer, and will show several examples of the results.

Characteristics Unique to z-Domain Curve Fitting

The most obvious characteristic of z-domain curve fitting is the use of periodic basis functions, resulting in a periodic frequency domain fit to the measured data. Thus, the basis functions should ideally have the same period as the original spectral data (or an integer multiple thereof), or else should span less than half of one period of the original data. This implies that the sampling rate for the curve fitter should ideally be some integer multiple of the digital data rate, so that an integer number of frequency response images occur within one period of the fitted result.

If some other fitter sample rate is required, then the x-cursors should be used to restrict the region of fit to no more than half of the data sampling rate. The fitter sample rate is often dictated by the system characteristics in which the resulting filter will be used, and thus may not be arbitrary. If the z-domain fitter is used on data obtained from an analog input, then the span of the analog spectrum should be less than half of the sampling rate of the fitter, or else the x-cursors should be used to restrict the fitting interval to be smaller than half of the fitter sample rate.

The span of an analog spectrum is given by the width of the display when in the 801 line mode (a 1024 line display mode can be selected under the WINDOW hard key). This 801 line boundary is indicated by vertical dashed lines when in the 1024 line mode. When these dashed lines are displayed, the fitting interval should be restricted to the interior region, due to the possibility of aliased components outside of this interval. This is done automatically by the curve fitter if x-cursors are turned off.

Time delays that are multiples of the sampling interval can be represented by poles at the origin in the z-domain. Thus, it is common to have multiple poles at the origin. Also, it is very convenient to cascade several identical digital filter sections together to obtain a composite transfer function, resulting in multiple poles or zeros at other locations, as well. Although multiple poles and zeros can also occur in the s-domain, it is relatively difficult to construct identical analog filter sections, so multiple poles and zeros are seldom measured in practice. Thus, multiple poles and/or zeros tend to be more common in the z-domain than in the s-domain.

Analog filters tend to be designed with more poles than zeros, or at least with some non-zero number of poles. However, all-zero digital filters are very common. All finite impulse response (FIR) filters comprise only zeros in their transfer functions. It is common to have more zeros than poles in a digital filter. This implies that the automatic polynomial order selection algorithm used in the curve fitter should behave differently in the z-domain than in the s-domain. Higher order filters tend to have more poles than zeros in the s-domain, but poles and zeros are treated equally in the z-domain.

Examples of z-Domain Curve Fitting in the HP 3563A Analyzer

A few examples will be given to illustrate the results that can be obtained from the curve fitter, and then some of the unique characteristics of the fitter in the HP 3563A analyzer will be discussed. The z-domain fitter works best when the measured frequency response is that of a digital filter having the same sampling rate as the one used by the fitting algorithm. However, it is possible to use a different effective sampling rate in the fitter, so that the same filter shape can be obtained based upon a new sampling rate. This might be useful if a new digital filter design having the same shape but a different sample rate were needed.

It is also possible to use the z-domain fitter on a measured frequency response from an analog filter. This allows a digital filter to be designed that matches an existing analog filter, at least out to half of the digital sampling rate. The sampling rate used in the fitting procedure determines the period of the orthogonal basis functions used in the fit. The following examples will show some of these curve fitting possibilities.

The digital filter used in these examples is part of a test accessory called the Hewlett-Packard ET 025379 signal processing subsystem, which is designed to train users in the use of the HP 3563A analyzer.

Figure 3 shows the magnitude of the frequency response of filter #1 (top trace), along with the output of the z-domain curve fitter (lower trace). The sampling rate of the actual digital filter is 7812.5 Hz, and this same rate is used in the fitting procedure.

Table 1 shows the poles and zeros calculated in the z-domain by the curve fitter. There is a pair of complex conjugate poles at a distance of 0.8992 from the origin, making an angle of ±17.945 degrees. The actual digital filter was designed with a pair of poles at a distance of 0.9 from the origin, making an angle of ±18 degrees. The measured zeros are at a distance of 0.9499 with an angle of ±35.890 degrees, and the correct results are a distance of 0.95 and angles of ±36 degrees. Refer to figure 2 for a view of the z-domain in which these poles and zeros can be located. Even though rectangular coordinates are given in the z-domain table, it is often convenient to visualize these numbers in polar coordinates, since the magnitudes of the poles must be less than unity and the natural frequency of each pole or zero is proportional to phase angle. Any poles that lie outside of the unit circle will be highlighted in the curve fit table.

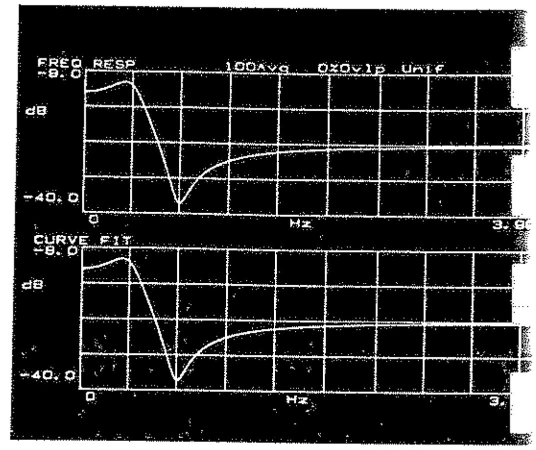


Figure 3: The measured frequency response magnitude of digital filter #1 (upper trace) having two poles and two zeros, and the result of the z-domain curve fitting algorithm (lower trace). The curve fitter sample rate is the same as the data sample rate. This example uses the full 1024 line interval for the fit, as well as for display.

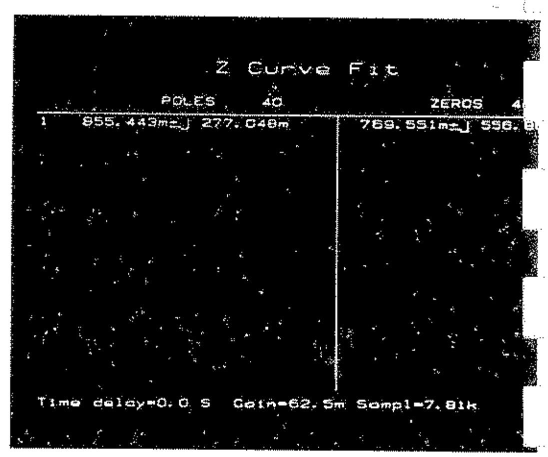


Table 1: The list of poles and zeros obtained in figure 3. Also note the gain constant and the curve fitter sampling rate.

Figure 4 shows the general state of the analyzer for linear frequency resolution measurements, and figure 5 shows the extra information needed to set up a digital measurement. Note that a digital burst random source was used to make these frequency response measurements, and that 100 source triggered averages were used. Digital quantization errors appear as distortion on the spectrum, but this can be converted to random noise and averaged to a smaller value by using the triggered burst random source and averaging over several measurements.

The above results were obtained using a curve fitter sampling rate that matched that of the measured data. In case the same filter shape is needed with a different sampling rate, the curve fitter can be used to refit the measured frequency response with the new sample rate specified. Figure 6 shows the fit to digital filter #1, using a curve fitter sample rate of 5000 Hz instead of 7812.5 Hz. Notice that both x-cursors are used to restrict the fitting interval to less than half of the digital sample rate. The markers show slight errors in the magnitude of the fit, which amount to about 0.1 dB in the filter passband.

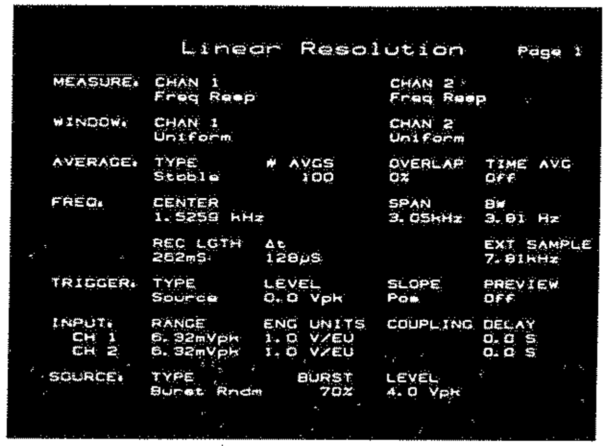


Figure 4:
The state of the analyzer for making linear frequency response measurements, using 100 triggered averages with a burst random source.

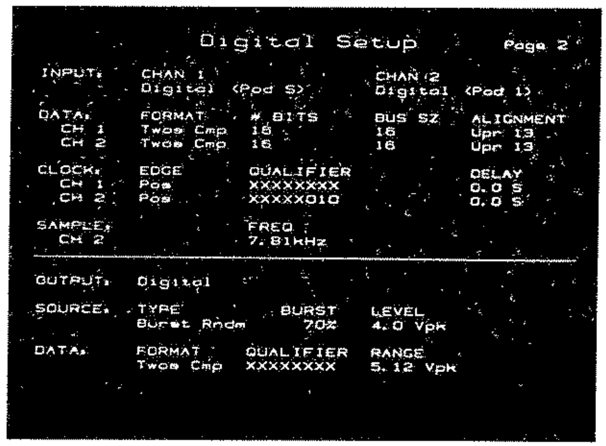


Figure 5:
Additional state information concerning the set up for digital inputs on both channels.

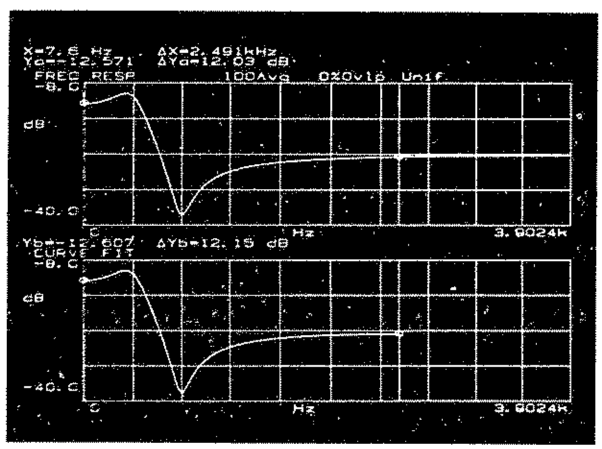


Figure 6:
The frequency response magnitude of digital filter #1 (upper trace) and the result of a z-domain curve fit using a lower sampling rate of 5 kHz (lower trace). Note the positions of the x-cursors in limiting the region of the fit.

Table 2 gives the new locations of the poles and zeros in the z-domain. Compare these values to those given in table 1. Since the fitter sample rate has been reduced, the angles of the poles and zeros have increased. The pole angle has increased from 18 degrees to about 28.2 degrees, and the zero angle has increased from 36 degrees to about 56 degrees. The magnitudes of these quantities have also changed somewhat.

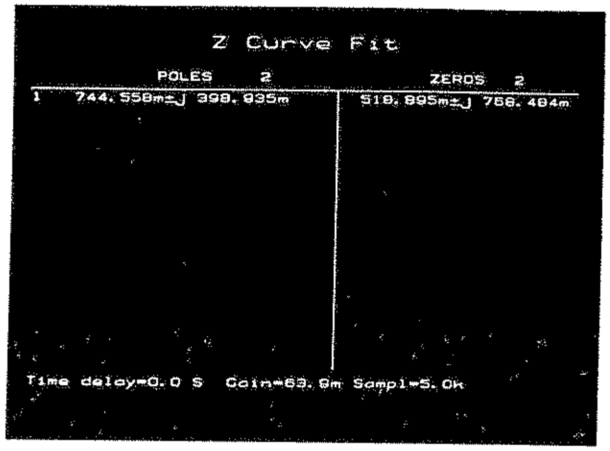


Table 2: The list of poles and zeros obtained in figure 6. The fitter sampling rate is shown.

Figure 7 shows the fit to filter #1 with a curve fitter sample rate of 10 kHz. The corresponding new poles and zeros are listed in table 3. In this case, the angles are smaller than those given in table 1 (where the sample rate was 7812.5 Hz). Note that x-cursors are not needed here, since the span of the data is less than half of the curve fitter sample rate (although these cursors can be used to further reduce the fitting interval if desired).

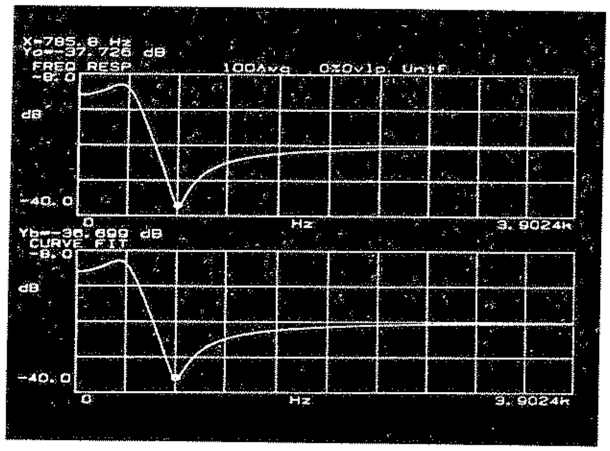


Figure 7: The frequency response magnitude of digital filter #1 (upper trace) and the result of using the z-domain curve fitter having a higher sampling rate of 10 kHz. Here, the entire frequency span is used for the fit (with x-cursors off).

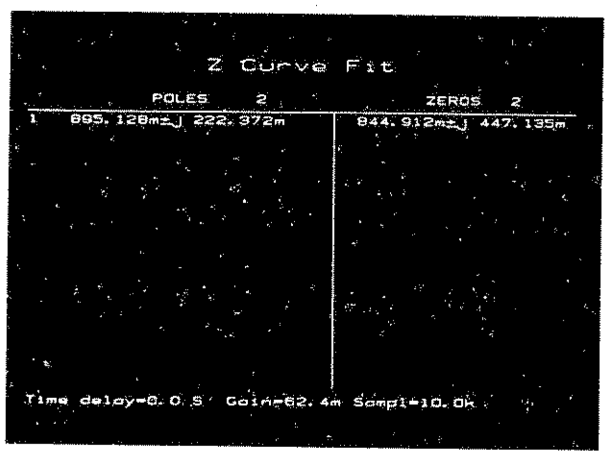


Table 3: The list of poles and zeros obtained in figure 7.

It is also possible to use the s-domain curve fitter in the HP 3563A analyzer to fit a z-domain frequency response function. This is a handy way to obtain an s-domain filter that matches a digital filter, for use in an equivalent s-domain model. This is one way to include digital components in an s-domain description of a control system, and might be used in making an s-domain root locus plot as a function of some loop parameter. Keep in mind that any aliasing components in the z-domain representation will introduce errors into the locations of the new s-domain zeros (and possibly poles as well).

Figure 8 shows this type of fit to the #1 digital filter response. The s-domain fit is fairly good except at the valley (789.6 Hz), where the error amounts to about 3 dB. Table 4 gives the resulting s-domain poles and zeros. If one additional pole is used in the fit, the valley point also fits well (with an error of only 0.16 dB). In general, do not use a single x-cursor during the fit, or else the fitting interval will only be ±20 data points around the cursor. Either turn x-cursors off, or else set them to the desired boundaries for the fit.

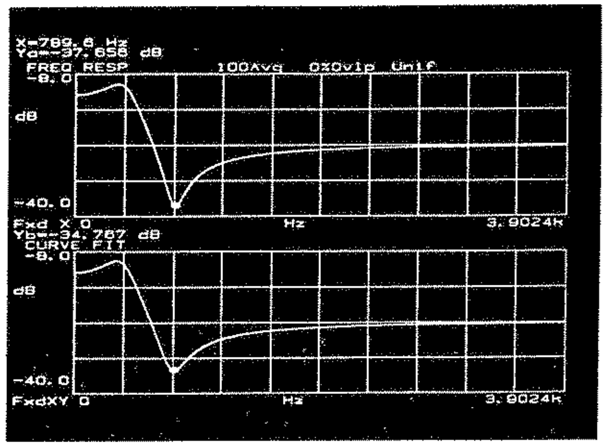


Figure 8: The frequency response magnitude of digital filter #1 (upper trace) and the result of using the s-domain curve fitter on this digital response function. Note that the valley fit is somewhat in error. One additional pole produces a good fit.

S Curve Fit

	POLES 2	ZEROS 2
1	-126.677 ±j 383.605	-90.2903 ±j 768.352

Time delay=0.0 S Gain=63.3m Scale=1.0

Table 4: The s-domain poles and zeros obtained in figure 8.

An analog filter is shown in figure 9, along with an s-domain fit to the magnitude, with x-cursors set to the range between 100 and 400 Hz. Figure 10 shows the passband of this filter, expanded. The resulting curve fit poles and zeros are listed in table 5. Notice that the three zeros were fixed at the origin during the fitting procedure (as indicated by the arrows after each fixed entry). This is accomplished by entering values of fixed zeros (or poles) into the curve fit table. Editing keys are provided for this purpose (CURVE FIT hard key and EDIT TABLE soft key). The values used in the original synthesis of this filter are given in table 6. The fitter results exactly match those used in the original synthesis.

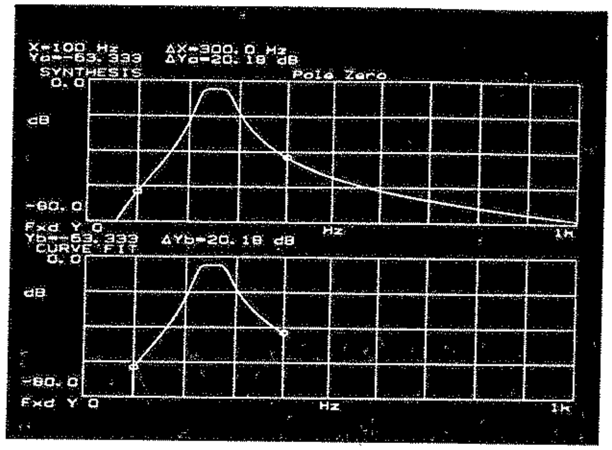


Figure 9: A synthesized analog filter response magnitude having six poles and three zeros (upper trace), and the result of using the s-domain curve fitter (lower trace).

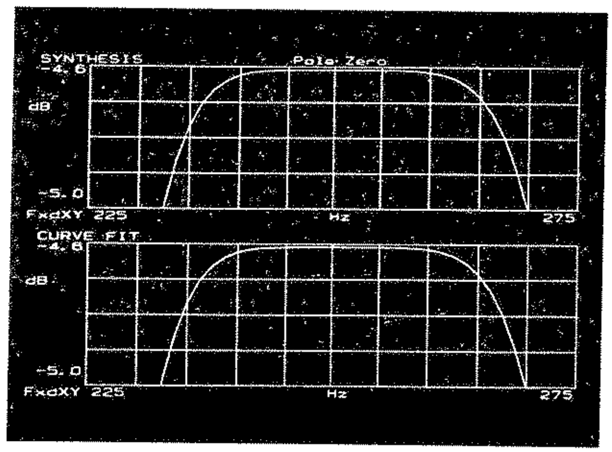


Figure 10: An expanded plot of the filter passband in figure 9, showing the accuracy of the s-domain fit.

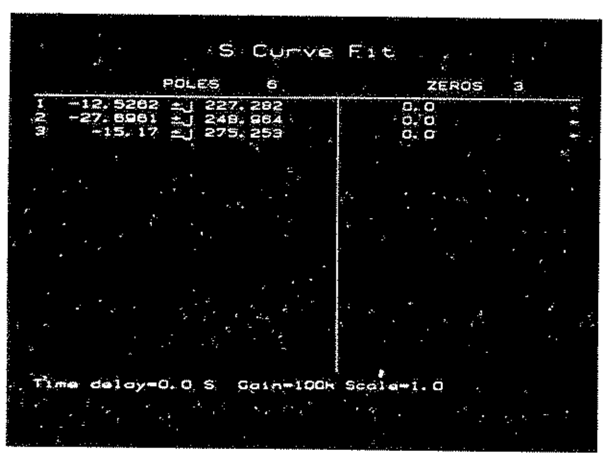


Table 5: The list of the measured s-domain poles and zeros for this analog filter of figure 9.

Next, the z-domain fitter is used to fit this analog filter shape. This would be of interest if a digital filter implementation of an existing analog filter were desired.

Figure 11 shows the results of the fit using a fitter sample rate of 2000 Hz. In this case, the markers are used to limit the fitting interval to a range between 100 and 400 Hz. The passband is shown expanded in figure 12. There is a small error of 0.008 dB in the passband magnitude. The curve fit table can be transferred to the synthesis table and a new trace can be constructed to show the fitted shape throughout the original frequency range.

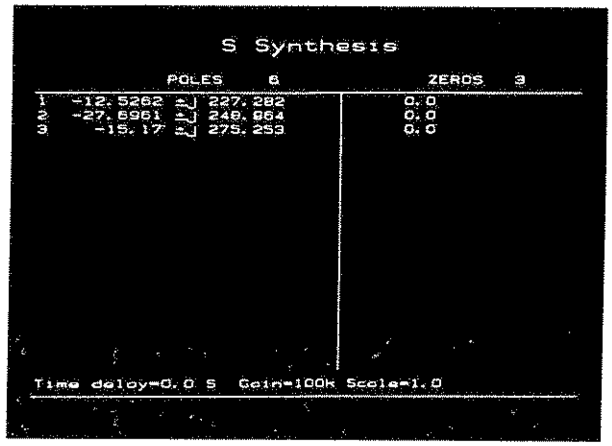


Table 6: The list of poles and zeros used to synthesize the upper trace of figure 9. Note that the curve fit results in table 5 are exactly the same.

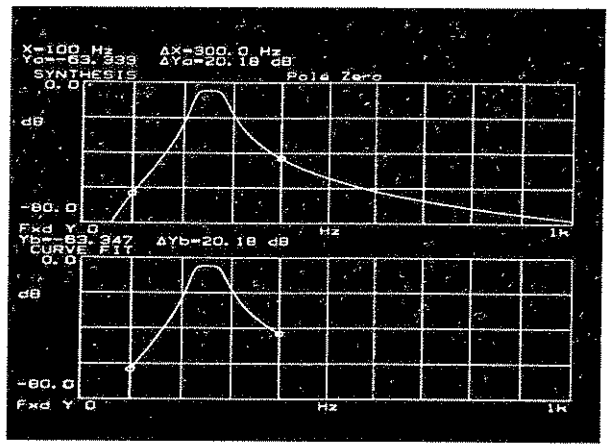


Figure 11: The upper trace is the same analog filter as shown in figure 9, but the lower trace is the result of using the z-domain fitter with a sample rate of 2 kHz. The fitting range is from 100 to 400 Hz.

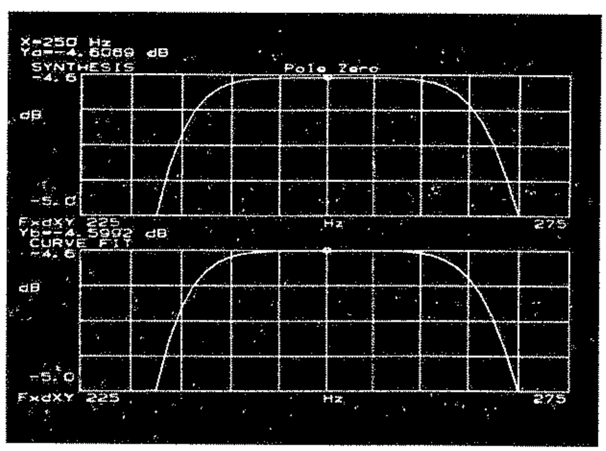


Figure 12: An expanded version of figure 11, showing the accuracy of the fit in the passband region.

The result of this z-domain synthesis is shown on the lower trace in figure 13.

The z-domain results of this fit to an analog filter are shown in table 7. The arrows indicate that three of the z-domain zeros were fixed at unity (by entering their values into the curve fit table) to account for the third order zero at the frequency origin. Also, the number of poles is limited to six, and only the number of zeros are allowed to change. This is a good way to prevent extra poles from appearing outside of the unit circle, while trying to compensate for the aliasing that has been introduced into the shape by sampling. The automatic order selection feature of the curve fitter can be used in this example by setting the maximum number of poles to six, and the maximum number of zeros to some sufficiently large number (up through 40). Press the CURVE FIT hard key, and the FIT FCTN soft key to show the AUTO ORDER soft key.

As the effective sampling rate used by the curve fitter is reduced, the fitting job becomes more difficult. In figure 14, the fitter sampling rate has been reduced to 900 Hz, which means that the negative image will alias into the band between 450 and 900 Hz. The fitting range is restricted to the region between 100 and 400 Hz.

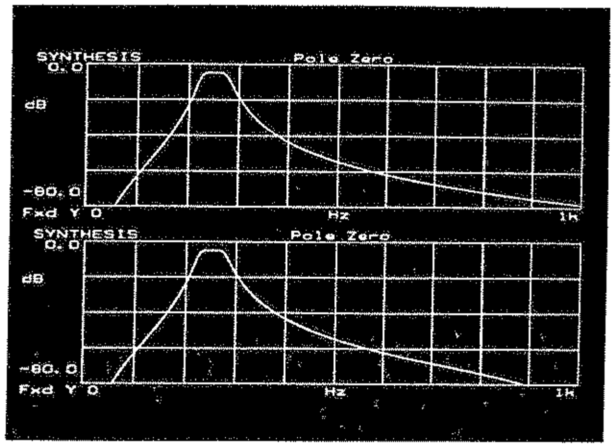


Figure 13:
If the curve fit table is transferred to the synthesis table, the complete curve fit trace can be reconstructed, as shown in the lower trace.

Z Curve Fit

	POLES 6	ZEROS 5	
1	726.581m±J 629.587m	-885.572m	
2	650.329m±J 648.148m	-3.03467m	
3	818.608m±J 725.51m	1.0	←
4		1.0	←
5		1.0	←

Time delay=0.0 S   Gain=1.18m   Sampl=2.0k

Table 7:
The list of z-domain poles and zeros that corresponds to the fit shown in figure 11. Arrows highlight the three zeros fixed before fitting, to account for the three zeros at the frequency origin in the original filter.

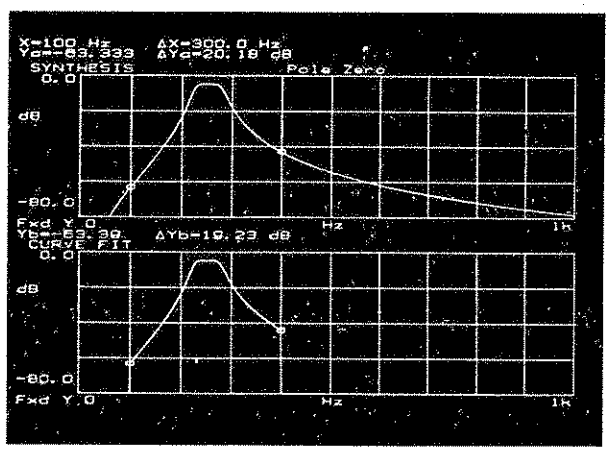


Figure 14:
The upper trace is the same analog filter as shown in figure 9, but the lower trace shows the result of using the z-domain fitter with a much lower sample rate of 900 Hz. The fitted region is still between 100 and 400 Hz.

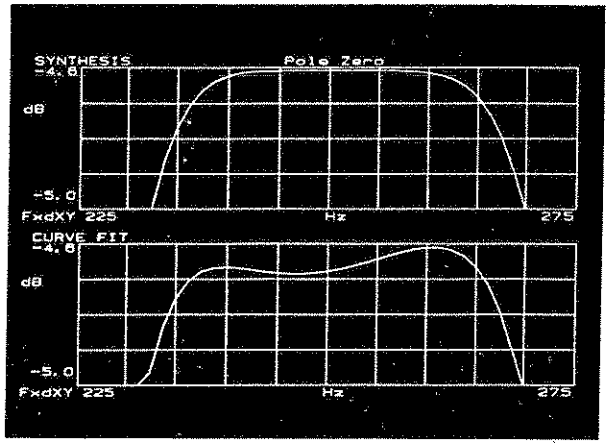


Figure 15:
An expanded version of figure 14, showing the accuracy of the fit in the passband.

Figure 15 shows an expanded view of the filter passband, and shows that the fitting error is less than 0.1 dB. When the fit is transferred to the synthesis table, the reconstructed fit throughout the original range is shown in figure 16. The presence of the aliased negative image is apparent. If the fitting interval had not been restricted, then the fitter would have attempted to fit with the shape shown on the lower trace. It is easy to predict that this would fail.

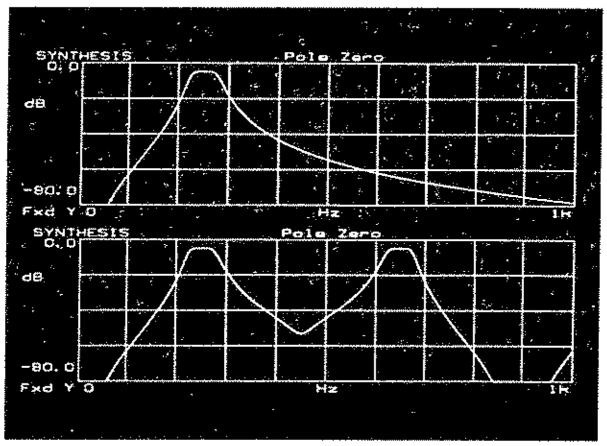
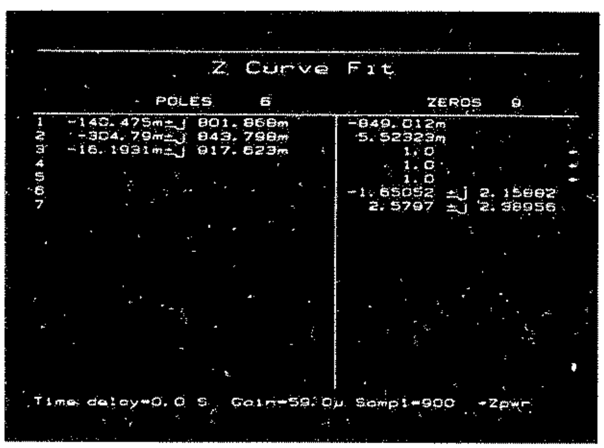


Figure 16:
The lower trace is a resynthesized shape obtained from the synthesis table after transfer from the curve fit table. Note the negative frequency passband region that has aliased into the original frequency range due to the low sampling frequency. This illustrates why the fitting region must be restricted to less than half of the curve fitter sample rate, when using the z-domain fitter.

Table 8 indicates the difficulty in fitting to an analog filter when the assumed sampling rate is too low. The number of poles has been held constant, and the three zeros at the frequency origin have been fixed. However, six additional zeros are required to compensate for the heavy amount of aliasing that is present in the digital representation. Many of these extra zeros are outside of the unit circle, indicating that the resulting digital filter does not have the minimum phase property, even though the original analog filter was of minimum phase form.


Z Curve Fit

	POLES 6	ZEROS 9
1	-140.475m±j 801.868m	-849.012m
2	-304.79m±j 843.798m	-5.52323m
3	-16.1931m±j 817.623m	1.0 ←
4		1.0 ←
5		1.0 ←
6		-1.65052 ±j 2.15882
7		2.5797 ±j 2.38956

Time delay=0.0 S Gain=59.0µ Smpl=900 +Zpwr

Table 8:
The list of z-domain poles and zeros that correspond to the fit shown in figure 14. The three zeros were fixed at the frequency origin as before, but there are six additional zeros in the list. These are needed to compensate for the aliasing that has occurred due to the low sampling rate.

Tips on Using the HP 3563A Curve Fitter

The general behavior of both the z-domain and the s-domain curve fitting algorithms in the HP 3563A analyzer is very similar to the s-domain curve fitter in the HP 3562A, except that the automatic order selection algorithm has been improved, and a new technique has been added to eliminate coincident pole/zero pairs after the fitting process is completed. These improvements tend to reduce the number of extraneous poles and zeros that are sometimes calculated. In addition, there are a few differences in using the HP 3563A analyzer (compared to the HP 3562A) that will be discussed below, along with a few tips on getting the best results from the curve fitter in the HP 3563A.

Select 801 or 1024 Line Display —

Under the WINDOW hard key, there is a softkey selection of either an 801 line or a 1024 line display. For analog measurements, the 801 line mode should be used, since the remainder of the data may be contaminated by aliasing components. However, there is no aliasing for all-digital measurements, so the 1024 line mode will often be needed. When the external digital sampling rate has been correctly specified, then the 1024 line display will show all frequencies up to (but excluding) half of the sampling rate. Note that the frequency span parameter always refers to the span in the 801 line mode.

The point at half of the sample rate (z=-1) is not displayed in the HP 3563A, even in the 1024 line mode. In addition, it is neither measured nor synthesized. Thus, the calculated impulse response of a digital filter will be missing this frequency component. This will often cause every other time point to alternate in sign around the correct mean value. To obtain the impulse response of a digital filter, use the PULSE soft key selection under the SOURCE hard key (after pressing SOURCE TYPE, and MORE TYPES), and measure the response in the time domain.

In the 1024 line display mode, when curve fitting zoomed data, or data where the span has been reduced enough for the digital anti-aliasing filters to be activated, there will be vertical dashed lines on the display screen, indicating the 801 line boundary. It is important to restrict the curve fitting region to the interior of these bounds to avoid contamination of the fit by aliased components. This is done automatically if the x-cursors are turned off.

Enter the Analog/Digital Sample Ratio —

For mixed domain measurements (one digital input and one analog input), an external analog sampling clock must be provided by the user, and the frequency must be a positive (non-zero) integer times the digital clock frequency (the maximum ratio is 512). This frequency ratio must be specified during the set up phase. Press the INPUT CONFIG (hard key), and set channel 1 (2) to digital and channel 2 (1) to analog. Then press the INTERFACE 1 (2) soft key, followed by SAMPLE CLOCK, and MIXED RATIO. Type the mixed ratio value on the numeric keypad, and press ENTER. The digital channel will be oversampled by the specified ratio, producing that number of images in the digital frequency response. The z-domain curve fitter will fit multiple images as long as the curve fitter sample rate is some integer multiple of the digital data sampling rate.

The selection of span values is limited to 100 kHz divided by 1, 2, 5 factors within each decade, when sampling internally in the HP 3563A. This same 1, 2, 5 sequence is also used to divide the external sampling frequency, whenever it is used. The available span values for digital or mixed domain measurements will be the digital sample rate divided by 2.56 and then divided by some member of this 1, 2, 5 number sequence. The factor of 2.56 allows for a factor of two to get half of the sample rate, and an additional factor of 1.28 to produce the 801 line span from the 1024 samples in the data block.

Limit the Fit Order —

If the optimum numbers of poles and zeros are known, the fitter works somewhat faster if these values are entered, and the USER ORDER mode is selected (CURVE FIT hard key and FIT FCTN soft key). However, it is often best to use the AUTO ORDER mode, in which the analyzer tries increasing orders until an adequate fit is obtained. The default maximum order values that are allowed are 40 poles and 40 zeros. These large numbers are seldom needed, and the fitter will work faster if these maximum values are set to smaller numbers (say 10 and 10).

When using the z-domain fitter to fit an analog frequency response function, it is usually best to keep the number of z-domain poles the same as the number of s-domain poles, if this number is known. Allow the number of z-domain zeros to climb as needed to compensate for the aliasing that is assumed by the fitter. This strategy reduces the probability that extraneous poles might fall outside of the unit circle in the z-plane.

Fix Known Pole/Zero Values —

If there are known values of any poles or zeros (such as poles or zeros at the frequency origin), then these should be fixed by entering them into the curve fit table before the fit is started. In the z-domain, if there are any poles or zeros at z=0, these can be fixed in the curve fit table or they can be removed and replaced by the equivalent time delay (or advance) before curve fitting begins. Poles and zeros can be fixed by pressing the EDIT TABLE soft key after the CURVE FIT hard key is pressed. Then the soft keys can be used to add new poles and/or zeros to the table, or existing poles and/or zeros can be fixed. Time delays or advances can be entered by pressing the TABLE FCTNS soft key and the TIME DELAY soft key, and then typing the desired delay via the numeric key pad.

When any z-domain poles or zeros are fixed in value, the period is assumed to be the curve fitter sample rate. This rate should match that of the measured data, since the effects of any fixed poles or zeros are removed from the original data before the curve fitting procedure is initiated.

Be careful not to confuse poles/zeros at the z-domain origin (representing time delays/advances) with poles/zeros at unity, representing poles/zeros at the frequency origin. This tends to be difficult to remember, since s-domain poles/zeros at the origin behave differently.

Set Digital Sample Rate and Curve Fitter Sample Rate —

When at least one of the signal inputs is digital, there will be a digital sampling clock that is provided by the user. In order to obtain the correct frequency axis scaling, it is important to accurately specify this external digital sampling rate (simply called the sample frequency) during instrument set up.

The z-domain curve fitter requires the selection of a curve fitter sample rate, which determines the period of all of the basis functions that are used in the fitting process. Press the CURVE FIT hard key, and then press EDIT TABLE, and TABLE FCTNS. This curve fitter sample rate does not need to be the same as the actual data sample rate, as long as there are no fixed poles or zeros, although the fit tends to be best when these two rates are equal. If the curve fitter sample rate is less than that of the digital data, then it is necessary to use the x-cursors to limit the fitting region to something less than half of the curve fitter sampling rate. If there are fixed poles or zeros, the curve fitter sample rate must be chosen to be the same as that of the data. The default choice is for the curve fitter sample rate to match that of the measured data.

Fitting Analog Data in the z-Domain —

As a rule of thumb when fitting to analog data, set the z-domain curve fitter sampling rate to around 10 times the highest frequency chosen for the fit, if this sort of choice is possible. The fit quality will be best when the fitter sampling rate is much higher than the fitting interval, but not so high that the filter passband is a very small portion of the interval of orthogonality of the basis functions used by the fitter (see reference [2] for more discussion).

If this rule of thumb is used, then try allowing two or three extra zeros to compensate for aliasing components. More zeros will be needed as the fitter sampling rate is reduced, relative to the fitting interval. It does not usually matter if the extra zeros are outside of the unit circle in the z-plane, although such z-domain filters will not have the minimum phase property.

Fitting With All-Poles or All-Zeros —

It is theoretically possible to fit any frequency response function with either all poles or all zeros. If the actual measured frequency response corresponds to either of these models, then the fitter will work fine using either poles only or zeros only. However, in general, it is very difficult to fit an arbitrary shape with either an all-pole or an all-zero filter, unless a very high polynomial order is used. In contrast, it is often relatively easy to fit an arbitrary shape with a relatively small number of poles and zeros combined. In the z-domain, any number of poles or zeros at the origin can be used to represent time delays or advances without violating the all-pole or all-zero assumption.

After curve fitting, it is possible that some poles may lie outside of the unit circle in the z-plane. signifying an unstable filter. This can often be corrected by adding extra zeros, and reducing the number of poles. The extra zeros may correct for excess phase in the data (due to aliasing), which otherwise can only be handled using poles outside of the unit circle. It is generally best to allow both poles and zeros in the fit, but to restrict the number of poles to a minimum, and to allow as many extra zeros as needed to give a good fit. The minimum number of poles may be known from the shape under consideration, or else it may be determined by a trial and error procedure.

Removing Time Delay —

If there is some time delay between the input and output signals from some device under test, this must be removed before curve fitting is attempted. This is true in both the s-domain and the z-domain, as well as for mixed domain measurements. However, time delays that are integer multiples of the curve fitter sampling interval Δt can also be removed by fixing that multiple number of poles (at $z=0$) in the z-domain curve fit table, if the z-domain curve fitter is being used. Otherwise, set the time delay (or advance) through the CURVE FIT hard key, and then EDIT TABLE, and TABLE FCTNS soft keys. Press TIME DELAY and then type the desired delay value on the numeric keypad and press ENTER.

Summary

This product note is a supplement to HP product note 3562A-3, entitled "Curve Fitting in the HP 3562A," which describes curve fitting in the s-plane to analog frequency response functions. In this note, the HP 3563A z-domain curve fitter is described for fitting to digital frequency response functions. In addition, this z-domain fitter can be used to fit analog responses, thereby obtaining an equivalent digital filter design that has the same frequency response function as an analog filter. It is also possible to fit a digital filter using a different sampling clock rate in the fitter.

The basis functions for the z-domain fitter are periodic complex exponentials, in contrast to the Chebyshev polynomials used in the s-domain fitter. All-digital frequency response functions are periodic, with the period equal to the sampling frequency, and the basis functions are chosen to match this periodicity.

Mixed mode measurements can be made where one of the inputs is a digital signal and the other is an analog signal. To make such measurements, two different sampling rates are often needed. The analog sample rate can be any non-zero positive integer times the digital sampling rate (up to 512). Either the s-domain or the z-domain curve fitter can be used on the resulting mixed frequency response functions, although extra poles and/or zeros may be needed in some cases to create a good fit. Refer to HP application note 243-4, entitled "Fundamentals of the z-Domain and Mixed Analog/Digital Measurements" for more discussion of mixed mode testing.

Several examples of fits to these various types of response functions are illustrated, and the resulting lists of poles and zeros are given. In addition, there is a list of tips that can help the user obtain good quality results from the curve fitting algorithm.

References

[1] Adcock, Jim, and Potter, Ron, "A Frequency Domain Curve Fitting Algorithm with Improved Accuracy," Proceedings of the 3rd International Modal Analysis Conference, Orlando, Florida, 1985, vol. 1, pp. 541-547.

[2] Product note HP 3562A-3, "Curve Fitting in the HP 3562A," HP part number 5952-0001.

[3] Application note 243-2, "Control System Development Using a Dynamic Signal Analyzer," HP part number 5953-5136.

[4] Application note 243-4, "Fundamentals of the z-Domain and Mixed Analog/Digital Measurements," HP part number 5952-7250.

HEWLETT PACKARD

For more information call your local HP sales office listed in your telephone directory or an HP regional office listed below for the location of your nearest sales office.

United States:
Hewlett-Packard Company
4 Choke Cherry Road
Rockville, MD 20850
(301) 670-4300

Hewlett-Packard Company
5201 Tollview Drive
Rolling Meadows, IL 60008
(312) 255-9800

Hewlett-Packard Company
5161 Lankershim Blvd.
No. Hollywood, CA 91601
(818) 505-5600

Hewlett-Packard Company
2015 South Park Place
Atlanta, GA 30339
(404) 955-1500

Canada:
Hewlett-Packard Ltd.
6877 Goreway Drive
Mississauga, Ontario L4V1M8
(416) 678-9430

Australia/New Zealand:
Hewlett-Packard Australia Ltd.
31-41 Joseph Street
Blackburn, Victoria 3130
Melbourne, Australia
(03) 895-2895

Europe/Africa/Middle East:
Hewlett-Packard S.A.
Central Mailing Department
P.O. Box 529
1180 AM Amstelveen
The Netherlands
(31) 20/547 9999

Far East:
Hewlett-Packard Asia Ltd.
22/F Bond Centre
West Tower
89 Queensway
Central, Hong Kong
(5) 8487777

Japan:
Yokogawa-Hewlett-Packard Ltd.
29-21, Takaido-Higashi 3-chome
Suginami-ku, Tokyo 168
(03) 331-6111

Latin America:
Latin American Region Headquarters
Monte Pelvoux Nbr. 111
Lomas de Chapultapec
11000 Mexico, D.F. Mexico
(905) 596-79-33

Data subject to change.

Printed in USA 9/89
5952-7251

HEWLETT PACKARD

Curve Fitting in the HP 3562A

Product Note HP 3562A-3

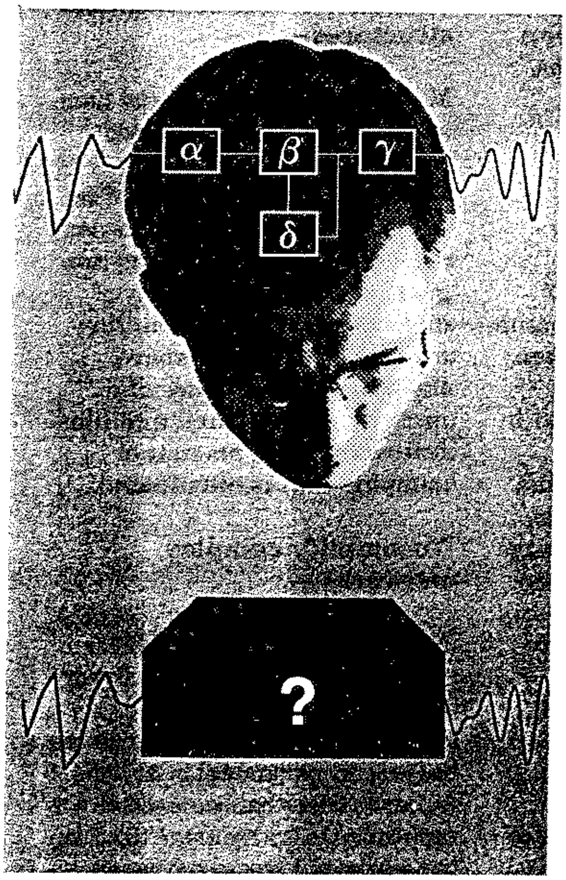


Figure 1: A measurement comprises the estimation of parameter values in a mental model of some physical system, using large amounts of "raw" input and output data collected from that system.

Introduction

This product note is intended to facilitate the use of the curve fitter in the HP 3562A dynamic signal analyzer. Curve fitting is used to estimate the s-plane parameters (poles and zeros) of a measured transfer function. The basic theory behind the curve fitting algorithm has been previously published in papers (see references [1] and [2] at the end of this product note), and most of the detailed instructions for operating the instrument are contained in the operating manual for the HP 3562A. However, curve fitting is still somewhat of an art, and there are many things that the user should know in order to obtain the best results. In most cases, a strict "cookbook" approach is not very practical. The bulk of this note is devoted to a review of the most important aspects of curve fitting beginning with a general discussion of transfer functions and curve fitters, and ending with a more specific listing of steps to take to obtain the best results from the HP 3562A.

Figure 1 illustrates the reduction of large amounts of "raw" data to a few simple numbers, or parameters, selected to describe the "information" contained in the original data set. When the raw data is collected from the inputs and outputs of some physical system, these parameters are called measurements.

In general, anyone making a measurement on a physical system has a mental model of the way the physical system works, and of the unknown parameter values that ultimately "match" the model inputs and outputs with those actually obtained from the physical system. This user wants an instrument that will estimate these model parameter values from the mass of raw data that is available.

There are two very important assumptions that are always made in this procedure. First, the results are not valid unless the assumed model is essentially correct. Second, there will be noise superimposed upon the raw data, so all estimated parameter values will also be noisy. The parameters are

Where is Curve Fitting Useful?

random variables that will never be quite "right," so there will always be some error in the results.

The behavior of many physical systems can be modeled by a set of linear differential equations (with constant coefficients) with respect to time. The solutions to these equations comprise a set of characteristic functions (complex exponentials), along with a set of characteristic parameter values (natural frequencies and damping coefficients).

One standard procedure for obtaining the values of these characteristic parameters is to calculate the Laplace transform of the differential equations and then to solve the resulting system of algebraic equations for the system variables. When the system input is a unit impulse, then the solution will be a rational fraction in the Laplace variable s , and is called the system transfer function. The roots of the numerator (zeros) and the denominator (poles), along with a constant gain factor, completely describe this solution.

A curve fitter is used to estimate poles, zeros, and gain factor from data that is collected at the input and output terminals of the physical system under test. The s-plane is then used as a catalog for these linear system parameters.

To refine models—

Figure 2 shows the closed loop frequency response function measured on a control system. Many designers think in terms of pole and zero locations in the s-plane when they are designing a control system, especially when they are working on loop compensation networks. Thus, good estimates of pole and zero locations are helpful in determining how closely the physical system agrees with the original design. See reference [3] for a discussion of control systems.

To "troubleshoot" design problems—

The curve fitter is also very useful to characterize the individual parts of a control system. For example, the actual object being controlled is often preceeded by some network (often called the "plant") and it is important to know the transfer characteristics of this component. The load on the control system may also change with frequency, so it can often be modeled as a separate component. Sometimes there are feedback sensors that involve amplifiers and filters, and thus have their own sets of poles and zeros. Finally, a special loop compensation network is generally needed to guarantee the stability of the loop under a variety of conditions.

To verify a design —

Figure 3 shows the frequency response of an elliptic filter. Lowpass filters of this type are generally built by cascading several stages, each of which comprises one or two poles, and some small number of zeros. The goal is to compare the measured parameters with those used in the original design.

To characterize mechanical structures—

Many mechanical systems have a number of lightly damped resonances that can cause various problems. These often account for acoustical emissions (such as squeals), and they can greatly affect the stability of any control loop of which they are a part. Figure 4 shows the frequency response of a mechanical structure, and illustrates the small amount of damping that is often found.

To simplify complex networks—

Sometimes a network (either mechanical or electrical) is very complicated and comprises a large number of poles and zeros, but a simpler model is desired. A curve fitter can be used to approximate a transfer function in a selected frequency band, and to obtain a reduced set of poles and zeros that still give an adequate representation of the physical system in that band.

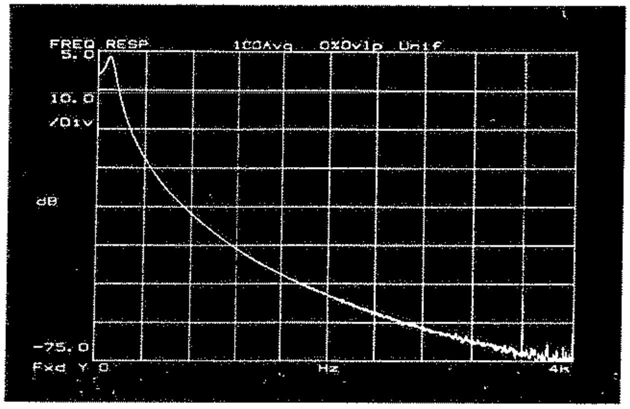


Figure 2: Closed-loop frequency response of control system

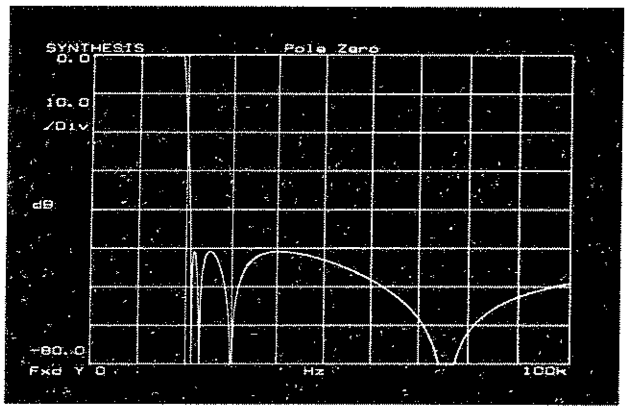


Figure 3: 8th order elliptic filter

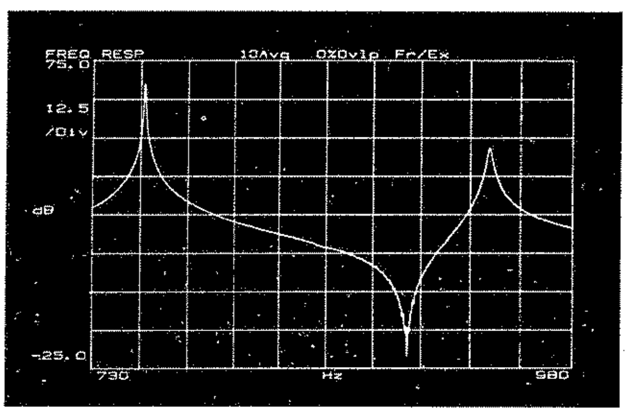


Figure 4: Resonances of rectangular plate

General Characteristics of Transfer Functions

Since the curve fitter in the HP 3562A is designed to estimate parameters of transfer functions in the s-domain, it is helpful to understand some of the general characteristics of such transfer functions.

The response of a linear physical system to an impulse can be expressed as a linear summation of contributions from each characteristic function of that system. The strength of each component is called the residue. Thus, a partial fraction representation of the transfer function, in which all components are summed together, is most intuitively related to the physical world. The independent physical parameters are pole locations and residue values.

When this partial fraction form is converted to a rational fraction form, the resulting numerator zeros are defined by a rather complicated combination of all of the poles and residues of the entire system. Thus, if the poles and residues are not determined correctly, then the zero locations will also be in error.

This triangular relationship between poles, zeros, and residues is both good news and bad news. A good curve fit will give correct values to all three sets of parameters, but any contamination of the original data will cause inconsistencies that prevent all three sets of parameter estimates from being simultaneously correct.

A frequency response function gives values of the transfer function along the imaginary (frequency) axis in the s-plane. The existence of the remainder of the s-plane can only be inferred. All we know is that "analytic" functions exist that can be described throughout the s-plane if their values are known only along the imaginary axis. Thus, we deduce the locations of s-plane poles and zeros from the measured frequency response data (via a curve fitting procedure).

Variations in pole and residue values have a small effect upon the data along the imaginary axis if the pole is a great distance from this axis. Conversely, any disturbances in the measured data will result in large uncertainties in pole locations and residue values for poles that are far from the imaginary axis. The same argument can be applied to zero locations. The net effect is that for any given amount of contamination on the measured data, it is easier to make accurate estimates of pole and zero values that are close to the imaginary axis than to estimate those values that are far away.

For the same reasons, it is very difficult to resolve the effects of a closely spaced cluster of poles and zeros, if they are located far from the imaginary axis. Within the noise level, many different cluster configurations can give essentially the same result along the frequency axis. Thus, it can become difficult or impossible to correctly determine the true cluster configuration.

By the same argument, if the exact nature of such clusters cannot be measured, then they can often be replaced with a simpler configuration that gives the same measurements on the imaginary axis. For example, if the curve fitter calculates a very closely spaced pole-zero pair, then (depending upon the level of noise) they can both be eliminated. It doesn't matter whether this pole-zero pair is in the left or right half of the s-plane. Similarly, roots that are far away from the frequency axis can often be dropped, as long as the gain constant for the transfer function is adjusted properly.

Even though linear systems are generally assumed, this is seldom completely true in practice. There are many types of distortion, but typically, the largest signal components generate the most distortion. However, the effects of this distortion are most pronounced on the small signal components. Thus, distortion components are often generated by signals in the passband of a filter (nearest the system poles), but the effects are most noticeable in the stopband (near the zeros). In addition, since stopband signals tend to be relatively small, the effects of noise and interference may be more apparent near the zero locations. These factors often make the estimates of zero locations more difficult than those of poles located at similar distances from the frequency axis.

General Characteristics of Curve Fitters

There are two basic types of data manipulation procedures: data transformation operations and data reduction operations. Transformations maintain the original information content, and hence are generally reversible. They simply rearrange the data to accentuate certain characteristics of interest. In contrast, data reduction algorithms are explicitly designed to reduce the original information content, and are based upon the assumption that the original data can be generated from a relatively simple mathematical model, with the addition of some sort of random noise or other external interference.

A curve fitting algorithm assumes a particular model and adjusts the internal parameters of that model so that the predicted output matches the actual output in some "best" manner. Often, a weighted sum of the squares of the errors between the original data and the fit is minimized. These are called weighted least-squares fitting algorithms.

There are several requirements that must be satisfied to produce good results from any curve fitting algorithm. For example:

(1) The mathematical model must be essentially correct or else results will be meaningless.

(2) Contamination of the data by noise and interference must be minimized. The estimated parameters will be random variables, but their variances will be determined by the amount of random noise on the original data. Also, non-random interference on the original data can introduce biases in the estimated parameter values.

(3) The resolution of the data samples along the frequency axis must be adequate to accurately represent both magnitude and phase information in the frequency response function. This is especially important when poles and/or zeros are very close together, or are near the imaginary axis. This is sometimes a problem at high frequencies when data samples are logarithmically spaced.

(4) The choice of weighting function that is applied to the fitting error is crucial in obtaining good results. Generally, data in regions having poor signal-to-noise ratio are de-emphasized, so fits tend to be poor in these regions.

(5) When some range of model parameter values all give very nearly the same degree of fit to the raw data, it is very difficult for the curve fitting algorithm to determine which combination of parameter values is the best. Thus, depending upon the noise on the original data, the selected parameter values can vary over a wide range, while still producing a "good" fit to the original data. Generally, some parameters are more sensitive to noise than others. Thus, good fits do not necessarily mean correct parameter values.

(6) If there are sources of interference or non-linear distortion, or if the data is contaminated in any way (such as by window leakage or aliasing), then it may be necessary to use a more complicated model to obtain a good quality fit. Subsequent interpretation of the results may be confusing, but the fault lies with "bad" original data. Generally, curve fitters work perfectly when the raw data is perfect, and the fit quality degrades as more contamination is added to the data.

(7) Even with clean data, the complexity of the required model is often unknown, so the fitting algorithm may need to try several orders of complexity before good results are obtained. For example, extra poles and zeros are often needed to obtain a good fit, but they subsequently tend to cancel one another. These extra roots act much like a catalyst in a chemical reaction. They enhance the fit and give better estimates of the actual root values, but then cancel themselves out of the final result.

(8) There are usually a number of out-of-band poles whose tails may be partially within the frequency band of the fitter. Extra poles and zeros will be needed to account for these tails, even though the out-of-band poles may not be accurately represented.

(9) Excess phase shift due to time delay may require a number of extra poles and/or zeros for an adequate representation. This delay will also tend to introduce errors in the true pole-zero values. Any known time delays should be removed before fitting is attempted.

(10) A curve fit is obtained by adjusting the coefficients on a set of "basis" functions until the best fit is found (to the measured transfer function). Even though the set of basis functions used in the curve fit may be mutually orthogonal, they lose that property when a weighting function is applied to the error. The measured frequency response data appears in the fitting equations as a second weighting function. The resulting weighted basis functions tend to lose much of their relative orthogonality if the measured data is concentrated in a narrow frequency band relative to the total band being fitted. In these cases, the fitting algorithm has trouble allocating the correct proportion of each basis function to the fit. This results in lower quality fits and in reduced tolerance to noise, interference, and distortion.

(11) It is best to break a response function into segments in which the data looks similar in each segment, and also spans most of the segment. For instance, it is generally best to fit a narrow passband separately, rather than to try to include a wide stopband.

(12) Fits are generally allowed to be poor in regions where the coherence is low (poor signal-to-noise ratio), on the theory that the best data should be used in the fit.

There is often a dichotomy between the user's view of the data, and the data actually used by the curve fitter.

The curve fitter uses data based upon a linear amplitude representation, whereas the user often prefers to view the data on a logarithmic vertical scale (say in dB). The curve fitter also uses both magnitude and phase (actually, real and imaginary parts) of the data, while users often judge the quality of fit based upon a display of magnitude only.

Actual curve fitting errors tend to have similar absolute magnitudes (depending upon the weighting function that is used), regardless of the magnitude of the spectrum in that region. Thus, relative fitting errors tend to be larger in the stopband region around the zeros, and this effect is greatly magnified by the logarithmic display.

The basis functions used in the curve fit (Chebyshev polynomials) can be sampled along the frequency axis with arbitrary spacing, as long as samples are close enough together to distinguish one basis function from another. However, when a logarithmic frequency scale is used (in either log resolution mode, or log swept sine mode), the frequency sample spacing in the high frequency region may be too coarse, relative to the basis function requirements. The quality of the fit degrades when this occurs.

Curve Fitting in the HP 3562A

It is perfectly acceptable to use logarithmic scales for displaying the curve fit results, but the behavior of the fitter is often easier to understand if the data is viewed using linear scales in both magnitude and frequency. Also, remember that phase is just as important as magnitude, as far as the fitter is concerned.

It should be apparent that even the best curve fitters cannot perform magic. When the raw data is clean, then curve fitting results are generally very good and are very repeatable. When data quality is poor, the curve fitter may need all the help that the user can give. Curve fitters work best with a sympathetic and cooperative user and cannot be considered to work automatically or infallibly.

The mathematical theory behind the curve fitting algorithm used in the HP 3562A is discussed in references [1] and [2], and will not be repeated here. Also, refer to the HP 3562A operating manual for operating details. The internal model assumes a transfer function that is a rational fraction (quotient of two polynomials) in the Laplace variable s . The results are either in terms of the coefficients of these polynomials, or are in terms of the roots (poles and zeros) of these polynomials, along with an overall gain constant.

A weighting function is automatically generated from the raw data and from the measured coherence function. The user can change this weighting function via front panel editing softkeys. Sometimes the quality of the fit can be improved by adjusting the shape of the weighting function.

An automatic polynomial order selection algorithm is included for cases where the appropriate orders are not known. This automatic mode can be overridden by fixed orders, whenever desired.

Known time delays in measured transfer functions should be removed before curve fitting is attempted.

The basis functions used in the HP 3562A curve fitter are Chebyshev polynomials, although the user is not aware of this during normal operation. However, a general knowledge of the nature of these functions can often help in understanding the behavior of the fitting algorithm. For example, at very low frequencies, all even order Chebyshev polynomials look like constants, and all odd order polynomials look like sloping straight lines through the frequency origin. Thus, if the data is restricted to this region, it is very difficult for the curve fitter to determine the optimum strength of each polynomial to produce the best fit to the data. Consequently, it is best to scale the measurement so that the data covers as much of the selected frequency span as possible.

The performance of the curve fitter on "clean" data can be readily tested by means of the synthesis capability in the HP 3562A. The frequency response function, resulting from any desired set of poles and zeros, can be synthesized in the instrument, and then the curve fitter can be applied to this data. The pole and zero values from the curve fitting algorithm can be directly compared to the original set.

Tables of poles and zeros can be transferred between the Curve Fit table and the Synthesis table. In the Synthesis mode, the transfer function can be converted between pole-zero form (times a gain constant), partial fraction form (pole-residue), and rational fraction form, comprising the coefficients of both numerator and denominator polynomials. Time delays can also be included, if desired.

Figure 8 shows the fit to the control system frequency response function illustrated in figure 2. This closed-loop response comprises a pair of complex conjugate poles and a single real pole, with all zeros at infinity. Table 4 lists the curve fit results. The fitter ignores the point at zero frequency so that dc offsets have no effect.

Figure 9 shows the fit to the frequency response of a mechanical structure (rectangular plate) excited by a hammer and monitored with an accelerometer. The corresponding coherence function and weighting function (generated automatically from the data) are shown in figure 10. Notice the poor coherence near the zero. The weighting function is designed to enhance the fits around both peaks and valleys, unless the valleys are too noisy.

The automatic order selection feature was used in both of these last two examples. For the control system, the correct orders were found. For the mechanical structure, the fits were never good enough for the order selection iteration to terminate. However, all fits involving an identical even number of poles and zeros above

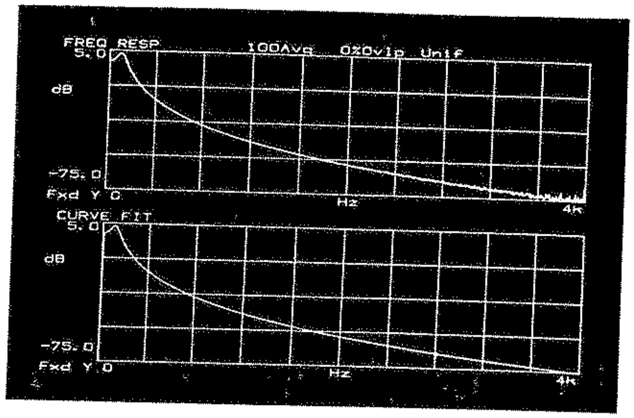


Figure 8: Curve fit to closed loop frequency response of control systems

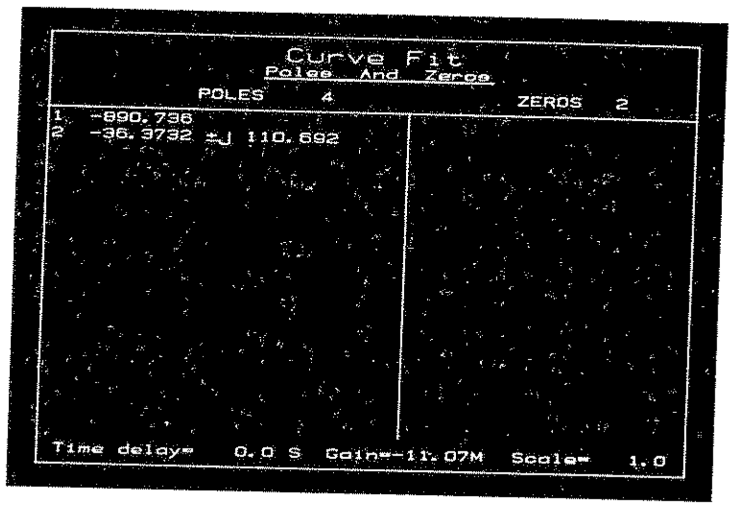


Table 4: Poles and zeros from curve fit to control system

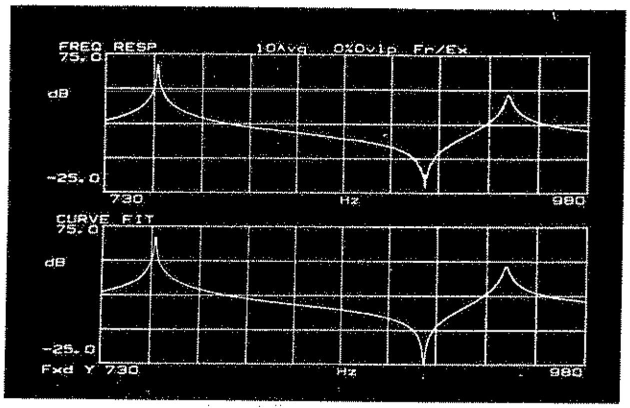


Figure 9: Curve fit to mechanical resonances in rectangular plate

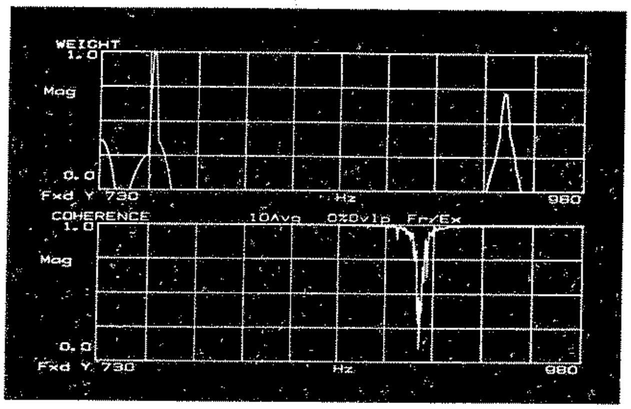


Figure 10: Weighting function and coherence function used in above fit

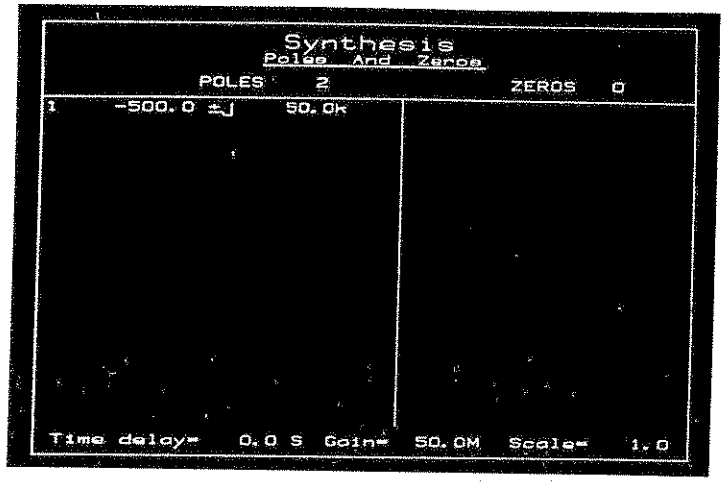
five seemed to give an adequate visual fit. Fixed orders of six poles and six zeros were selected for the final plot in figure 9. Even if the auto-order procedure does not stop, it is still possible to watch its many tries and to select suitable orders for subsequent use. There is often a preference for either even or odd numbers of poles and/or zeros.

Even though the fit may be poor for one particular choice of orders, it may be very good for the next choice (and poor for the next!). There will generally be a minimum order, below which none of the fits will be good. Then there will be a range where fits tend to be reasonably good. Finally, for very high orders, the fitter will begin to try to fit noise, etc., and the fit quality will begin to deteriorate. Whenever a data reduction algorithm is used, there is generally a fine line between estimating too few parameters, and estimating too many.

The effects of time delay on the original data is illustrated in tables 5 through 8, and figures 11 and 12. Table 5 lists the parameters of a two pole synthesized frequency response function. Figure 11 shows the phase plot of both the synthesized function and the subsequent curve fit to that function. Time delay is zero, so the phase simply changes by 180 degrees in the negative direction as each pole is passed in the positive frequency direction.

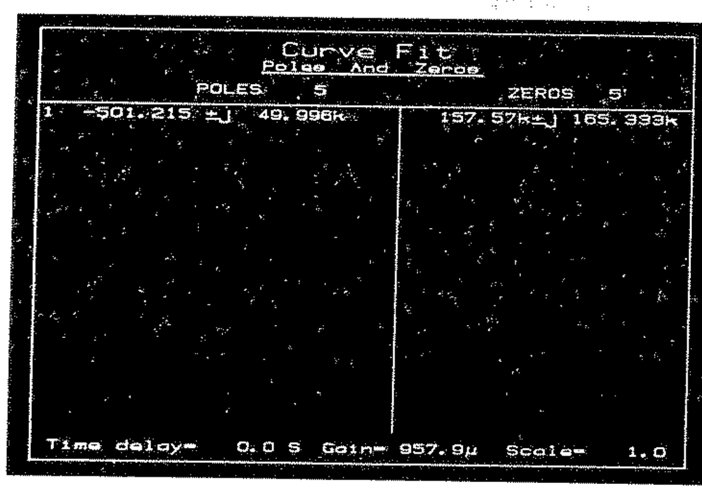
When 1 microsecond of delay is introduced into the synthesized shape without informing the curve fitter, an extra pair of right half-plane zeros is added by the curve fitter (table 6) to compensate for the extra phase. When 10 microseconds of delay are introduced, table 7 shows the extra roots that are added. In this case there are four extra zeros in the right half-plane and two extra poles in the left half-plane.

However, when the curve fitter is given a time delay of 10 microseconds, the fit only requires the original pair of poles, as indicated in table 8. The phase plots for this case are illustrated in figure 12. Note the negative phase slope resulting from the time delay.



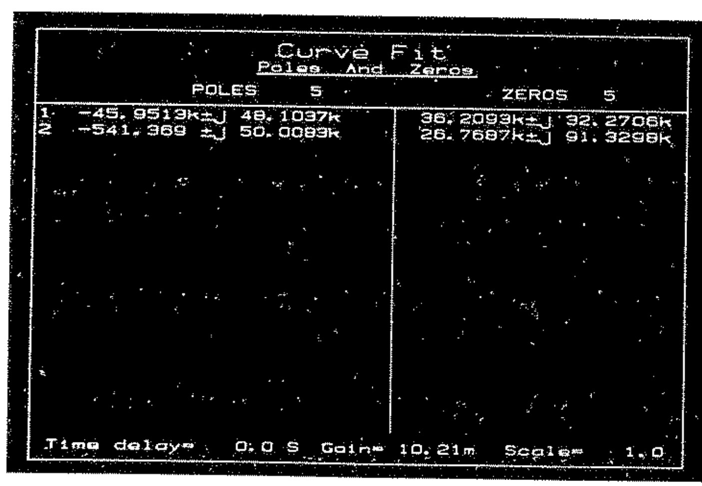
Synthesis
Poles And Zeros
POLES 2 ZEROS 0
1 -500.0 ±J 50.0k
Time delay= 0.0 S Gain= 50.0M Scale= 1.0

Table 5: Synthesized pole pair without time delay



Curve Fit
Poles And Zeros
POLES 5 ZEROS 5
1 -501.215 ±J 49.996k | 157.57k±J 165.333k
Time delay= 0.0 S Gain= 957.9μ Scale= 1.0

Table 6: Curve fit results for pole pair having 1 microsecond of delay



Curve Fit
Poles And Zeros
POLES 5 ZEROS 5
1 -45.9513k±J 48.1037k | 36.2093k±J 32.2706k
2 -541.369 ±J 50.0083k | 26.7687k±J 91.3298k
Time delay= 0.0 S Gain= 10.21m Scale= 1.0

Table 7: Curve fit results for pole pair having 10 microseconds of delay

Curve Fitting Check List for the HP 3562A

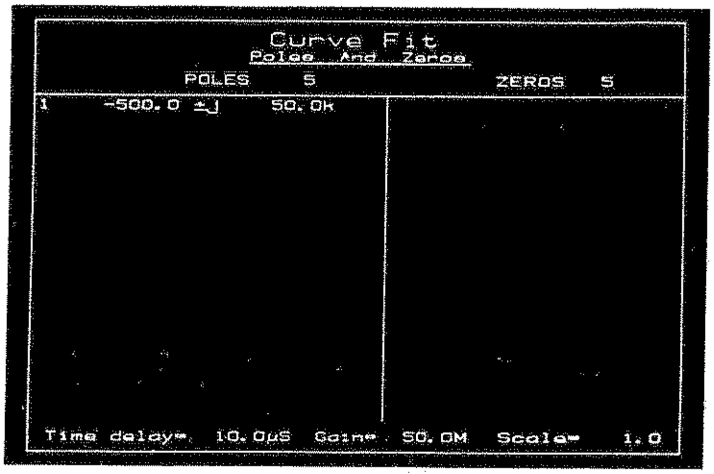


Table 8: Curve fit results for pole pair with 10 microseconds of delay removed prior to fitting

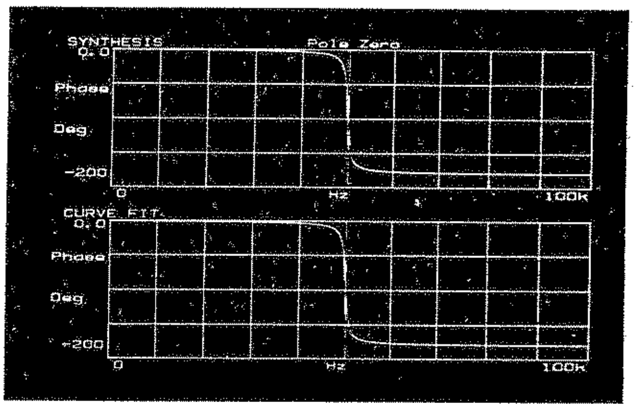


Figure 11: Phase of curve fit to pole pairs without time delay

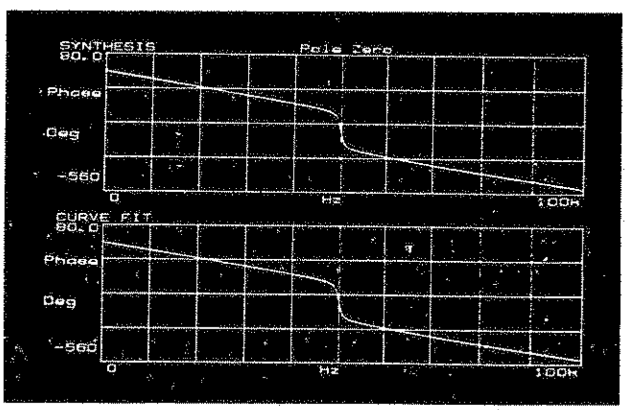


Figure 12: Phase of curve fit to pole pair with 10 micro-seconds of delay removed prior to fitting

To obtain the best curve fit results, follow the procedures below:

(1) Start by making the best frequency response measurement possible. This is very important.

(a) Select a source type and time window to minimize leakage and distortion. Use a Burst Chirp or Burst Random source with a Uniform (rectangular) window, or a hammer with a Force-Exponential window, or Swept Sine. Adjust source magnitude for the most linear operating region.

(b) In all but the Swept Sine mode, use as many averages as possible. Ideally, the coherence function should be nearly unity at all frequencies of interest.

(c) Choose the frequency span to cover the smallest range of interest. Make separate measurements on each region of interest, if possible. Avoid cases where the band of interest is in a very narrow region of the chosen span, if at all possible. Also, avoid large zoom factors. Baseband measurements are easiest to fit.

(2) Unless previous experience indicates otherwise, use the Auto Weighting function and the Auto Order selection mode. Then watch the fit quality and the corresponding orders. Note the orders that give the best fit, and then specify these in the User Order mode for a closer inspection.

(3) Always allow a few more poles and zeros than the expected number, to help take care of tails from out-of-band poles, and from other sources of noise, distortion, and interference. These extra roots will often cancel one another in the curve fit table. The valid roots are usually fairly obvious by comparing the table entries to the original data. Don't be concerned about right half-plane poles if there are matching right half-plane zeros. If a pole is removed, the gain constant must be readjusted by dividing by the negative of the pole value. Likewise, if a zero is removed, multiply the gain constant by the negative of the zero value. If matching pole-zero pairs are removed, the gain constant is not affected. If poles or zeros are removed, it is good practice to resynthesize the fit, to be sure that it is still acceptable (transfer the Curve fit table to the Synthesis table and adjust the gain constant, if necessary).

(4) It is important to specify any pure time delay that exists in the measured data, so that the curve fitter can remove this delay before attempting a fit. Otherwise, it may be difficult to obtain a good fit, even with a large number of poles and zeros. Excess phase due to right half-plane zeros should not cause any fitting problems, as long as enough zeros are allowed in the order selection step.

(5) Occasionally, the weighting function can be modified by the user to improve the quality of the fit. However, the fit does not always improve in regions where the weighting function is increased. Generally, the cleanest data should be weighted the most, and this is often near the peaks of the frequency response function.

(6) If some of the pole and/or zero values are known, like those at the origin, for example, then those should be prespecified and fixed. However, be certain that they are correct, or else the fitter will be confused.

(7) It is possible for the fit to be good, but for some of the poles and/or zeros to be incorrect. This can happen if more than one pole-zero configuration gives the same frequency response function, within the limits of the noise level.

(8) It is necessary to have sufficient frequency resolution to adequately represent the true response function. The frequency resolution should be less than the reciprocal of the time duration of the impulse response of the system under test. Thus, the minimum spacing between poles and zeros should be several frequency sampling intervals, or else the fitter does not have adequate information for a good fit. The solution is either to make a cleaner measurement, or to reduce the frequency span to improve frequency resolution.

(9) In order to reduce the range of numbers that must be handled in the processor, it is best to choose a frequency scale factor so that frequency numbers are near unity. This is not a problem for low order fits, but can become important if the orders are very large. For example, if the span of interest ranges between 10 Hz and 100 kHz, then set the frequency scale factor to 1 kHz. The resulting scaled frequency numbers will range between .01 and 100.

Summary and Conclusions

There are a few HP 3562A operating details that should be emphasized, to help the first-time user:

(1) When generating a synthesized trace from a table of poles and zeros, the gain constant should be selected so that the passband gain is near unity. This helps keep the range of numbers under control in the curve fitting step.

(2) In the curve fit mode, using synthesized data, choose A & B traces for the fit, where the synthesized trace is in A. The contents of trace B are not actually used in this case. Also, a weighting function of unity is assumed at all frequencies.

(3) If the Auto Order mode is selected, any non-zero entries represent the maximum orders that the curve fitter is allowed to use. However, a zero order value is not an upper limit, so the order is allowed to increase, up to 40.

(4) When the x-cursor is turned off, the right-most boundary of the curve fitting interval is automatically set to where the weighting function drops (and remains) below .001. This choice can be overridden by using two x-cursors to define the desired frequency interval. A single x-cursor specifies a fitting interval that is only ±20 frequency bins in width. In all cases, the Chebyshev polynomial basis functions are scaled to span the range indicated by the right-most frequency bin that is used in the fit.

The curve fitter in the HP 3562A is used to estimate the coefficients of a rational fraction representation (in the s-plane) of a measured frequency response function. The resulting poles and zeros of the transfer function can be readily obtained from these coefficients.

For best results, the original measured data must be as "clean" as possible. Any contamination by noise, distortion, or interference will degrade the result. In addition, any known time delay must be removed before attempting a fit.

An error weighting function is automatically calculated (although the user can edit this function), and an automatic order selection algorithm is included. The user can set the orders manually, but extra poles and zeros should be allowed, to facilitate the fit to out-of-band tails, and to various sources of distortion.

There are numerous factors that can cause the fitter some degree of difficulty and the user should be aware of these. In general, the fitting algorithm works very well, but there is no magic. The fitter should not be used blindly, or in a completely automatic mode.

References

[1] Adcock, J., and Potter, R., "A Frequency Domain Curve Fitting Algorithm with Improved Accuracy," Proc. of 3rd International Modal Analysis Conf., 1985, Orlando, Florida, pp. 541-547.

[2] Adcock, James, "Curve Fitter for Pole-Zero Analysis," Hewlett-Packard Journal, Jan., 1987, pp. 33-36.

[3] "Control System Development Using Dynamic Signal Analyzers," Hewlett-Packard Application Note 243-2, 1984.

Glossary

Aliasing: When a time record is sampled at regular intervals, any frequencies above half of the sampling rate are converted to lower frequencies. This frequency conversion process is called aliasing.

Basis functions: These comprise a set of functions into which any arbitrary function can be decomposed. Thus, any arbitrary function can be represented as the sum of these basis functions, each multiplied by some suitable coefficient.

Characteristic functions: A class of functions that are solutions to some differential equation.

Characteristic values: Parameters that determine the exact characteristic function to be used, out of the entire class of these functions.

HEWLETT PACKARD

Chebyshev polynomials: A set of polynomials that are mutually orthogonal over the real interval (-1,1).

Coherence: The coherence between an output signal and an input signal is that proportion of the total output power that can be attributed to the input signal.

Compensation network: In order to stabilize a control system (so that it does not oscillate), it is often necessary to insert a network into the control loop to adjust the loop gain and phase margins.

Curve fitting: The adjustment of the parameters of a mathematical model of a physical system, so the performance of the model matches the measured performance of the physical system in some optimum manner.

Frequency response function: A transfer function evaluated along the frequency (or imaginary) axis in the s-plane.

Multiple pole: A root of the denominator of a transfer function that appears multiple times.

Orthogonality: Two functions are orthogonal if the integral of their product is zero, over some interval.

Partial fraction: The sum of terms, each of which comprises either a coefficient times a power of the independent variable, or a coefficient (called the residue) divided by a monomial in the independent variable. The monomial root is called a pole.

Poles: Roots of the denominator polynomial of a transfer function.

Rational fraction: The quotient of two polynomials.

Residue: A coefficient multiplying each term in a partial fraction representation of a pole of a transfer function.

s-plane: s is the independent variable in the Laplace transform of a time waveform. The s-plane is a representation of this complex variable.

Transfer function: An s-plane representation of the relation between the input and the output of a linear system. It can be represented as the quotient of two polynomials, or in partial fraction form.

Weighting function: A function along the frequency axis that "weights" the error in the curve fit, so that some regions are given more influence than others in the final quality of the fit.

Window leakage: When a time record is multiplied by a time window, the frequency spectrum of the original signal is "smeared," and new sidebands appear around the original signal. This smearing process is called leakage.

For more information call your local HP sales office listed in your telephone directory or an HP regional office listed below for the location of your nearest sales office

United States:
Hewlett-Packard Company
4 Choke Cherry Road
Rockville, MD 20850
(301) 670-4300

Hewlett-Packard Company
5201 Tollview Drive
Rolling Meadows, IL 60008
(312) 255-9800

Hewlett-Packard Company
5161 Lankershim Blvd.
No. Hollywood, CA 91601
(818) 505-5600

Hewlett-Packard Company
2015 South Park Place
Atlanta, GA 30339
(404) 955-1500

Canada:
Hewlett-Packard Ltd.
6877 Goreway Drive
Mississauga, Ontario L4V1M8
(416) 678-9430

Australia/New Zealand:
Hewlett-Packard Australia Ltd.
31-41 Joseph Street
Blackburn, Victoria 3130
Melbourne, Australia
(03) 895-2895

Europe/Africa/Middle East:
Hewlett-Packard S.A.
Central Mailing Department
P.O. Box 529
1180 AM Amstelveen
The Netherlands
(31) 20/547 9999

Far East:
Hewlett-Packard Asia Ltd.
22/F Bond Centre
West Tower
89 Queensway
Central, Hong Kong
(5) 8487777

Japan:
Yokogawa-Hewlett-Packard Ltd.
29-21, Takaido-Higashi 3-chome
Suginami-ku, Tokyo 168
(03) 331-6111

Latin America:
Latin American Region Headquarters
Monte Pelvoux Nbr. 111
Lomas de Chapultepec
11000 Mexico, D.F. Mexico
(905) 596-79-33

Data subject to change.

Printed in U.S.A. 4/89
5952-0001