Signalyze™

Signal Analysis for Speech and Sound

User's Manual

by Eric Keller, Ph.D.

Version 3.0 August 1994

InfoSignal™ *Inc.*

www.infosignal.com

Grateful acknowledgments to:
Stefan Werner, Joensuu, Finland
Philip Keller, Seattle, Washington
Pascal Perrier, Grenoble, France
Lorenzo Morellini, Lausanne, Switzerland
Brigitte Zellner, Lausanne, Switzerland
Feng Gang, Grenoble, France
Philippe Martin, Toronto, Ontario

Note
Signalyze is constantly being revised and improved.
Your reports of bugs and inconveniences are highly appreciated.

Signalyze incorporates portions of code from/by:
Invention Software, Ann Arbor, Michigan
Digidesign, Menlo Park, California
GWInstruments, Somerville, Massachusetts
The "Sid Trio" (Eric Gould, Jeffrey Siegel and Dave Fleck)
Joe Pillera, Ann Arbor, Michigan
Chris Johnson, Dallas, Texas
Michael S. Engber, Illinois
All external code is used with permission of the respective copyright holders.

Signalyze is written in Think C.

Trademarks: Signalyze, InfoSignal and the brain-signal icon are trademarks of InfoSignal Incorporated. InfoSignal Inc. is a corporation established under the Federal Laws of Canada. It maintains a representation in Switzerland.

Other trademarks: Macintosh, Mac and MultiFinder -- Apple Computer; Microsoft -- Microsoft Corporation; StatView -- Abacus Concepts, Inc.; SYSTAT -- SYSTAT, Inc.; MacADIOS and MacSpeech Lab -- GW Instruments, Inc.; PhotoShop -- Adobe Systems Inc.; Think C -- Think Technologies, Inc.; Farallon™, MacRecorder® -- Farallon Computing, Inc.; SoundWave -- Impulse Inc.; CompuServe -- CompuServe; Ultrapaint -- Deneba Systems Inc.

The Signalyze brain-signal icon is a trademark of InfoSignal Inc.

Manual and Software © Copyright 1988-1994 by Eric Keller

Contents

Foreword 5

A Summary of Version 3.0 Additions 7

What is Signal Analysis? 17

What Signalyze can do for you 18

Welcome to Signalyze 20

A Guided Tour of Signalyze 21
- Tutorial 1: Signalyze Basics 22
- Tutorial 2: First Steps 28
- Tutorial 3: Sound Reproduction and Sound Input 32
- Tutorial 4: Manual Scoring 38
- Tutorial 5: Spectral Analysis 54
- Tutorial 6: Multiple Signals and Signal Editing 69
- Tutorial 7: Transformations, amplitude measurement and filtering 76
- Tutorial 8: Advanced Operation 87

Installation Suggestions 104
- Recommended Hardware 104
- Signalyze Installation 105
- What do you do if you don't have enough memory? 107
- User Support 107
- Updates 108

User's Reference Section 109
- Buttons: Signalyze Keyboard and Clicking Actions 109
- Buttons: The General Control Panel 113
- Buttons: The Signal Control Panel 116
- Scroll Bars 119
- Menus 120
  - Apple menus 120
  - File menus 122
  - Edit menus 132
  - Signal menus 152
  - Spectral menus 164
  - Window menus 184
  - Transform menus 187
  - Macro menus 204

Technical Information on Signalyze 210
Ergonomic Considerations 210
Memory Management Considerations 214
Signal File Formats 216
Label File Format 225
Numeric Formats 226
The Signalyze Text Editors 227

The Most Common Questions about Signalyze 228

License Agreement 232

References 236

Appendix 236

Index 242

Foreword

With 3.0, Signalyze™ enters its fifth official year of existence. Version 1.0 was launched in January 1990, Version 2.0 came out in January 1992, and this version 3.0 is officially released in August 1994.

Seasoned users of Signalyze will immediately notice some differences. A new tool palette appears at the bottom of the signal window, the interface has a snazzy new embossed look—and so do the new Signalyze icons. Behind the fresh new interface, there are some substantial additions and improvements. Most importantly, we now have an extensive and user-friendly labeling facility. Not only can you mark points and sections in the signal, you can also set labeling levels, label colors and label fonts. Once you have a set of labels, you can save them as text files, you can edit them, delete them, or search them by level or by contents. Manual scoring has just become a great deal easier.

Another useful addition is the speech slow-down feature. When segmenting continuous speech, it is often difficult to tell which vowel one's looking at. But now that we can slow down speech signals up to five times, identifying sounds has become a cinch. Also of great help is the new vertical zoom setting. Choose zooms between 0.5 and 64 times the signal height. At large zooms, you can easily see all the detailed features of a signal's quiet portions, and at reduced zooms, you can display the signal without interference from the time labels.

Other new features may appear minor, but to some users, they're very important: There is support for the new AV- and Power PC-machines, there are color- and grayscale spectrograms right on the signal window, spectrograms stay around between card changes, scale limits for big plots can be set manually, there's a routine for automatically trimming edges of signals, and there are separate menu items for resetting the signal start and the signal's minimum and maximum values.

More good news will follow over the course of the next year. Work is in progress to make Signalyze fully programmable. By end of 1994, a supplement to this manual and an upgrade to Signalyze will come out that shows you how to script the Signalyze functions both from inside and outside of the program. The numerous French-language users of the program can now get a French version of this manual. More analysis functions will be added. And many more minor improvements will follow in due course—some of them in response to your numerous personal requests.

So the most important message to the user is: Stay tuned! Signalyze is a continuously evolving program. Over the last two years, there have been 51 upgrades to Version 2.0. On the average, that's an upgrade *every*

second week. Some of these were "simple" bug fixes. Some were adaptations required by the appearance of new Macintoshes that have new features and new particularities[1]. And quite a few of these upgrades were "freebies"—useful improvements that were implemented without major changes to the interface.

How do you keep up with all these changes? Easy for those who have access to FTP or Gopher services (that's the majority of Signalyze users). Simply get the latest updater and update notice from one of the two servers. Check out the User Support section of this manual and the latest InfoSignal Flash news for details on how to gain access to these servers.

It's nearly as easy for those who have *no* access to servers: About every half year, or whenever you encounter a problem, put a diskette into a self-addressed envelope and send it to the Lausanne InfoSignal Inc. office. You'll get the latest updater in exchange for your efforts. Please do understand that with the growing number of Signalyze users, and in view of the fact that Signalyze is still largely a one-man operation, sending out free upgrade diskettes to everyone has become increasingly difficult.

What hasn't changed in all these years is that I still answer all your email questions—personally, as thoroughly, and as fast as I can. I get about three to five questions a day. I enjoy this personal contact, and I try to accommodate as many of your personal requests as possible. It tells me much about what's good in Signalyze, and what needs fixing. In that sense, Signalyze is bound to remain a special program for a while to come. It's a program where *you* can still make a difference.

With that, happy Signalyzing!

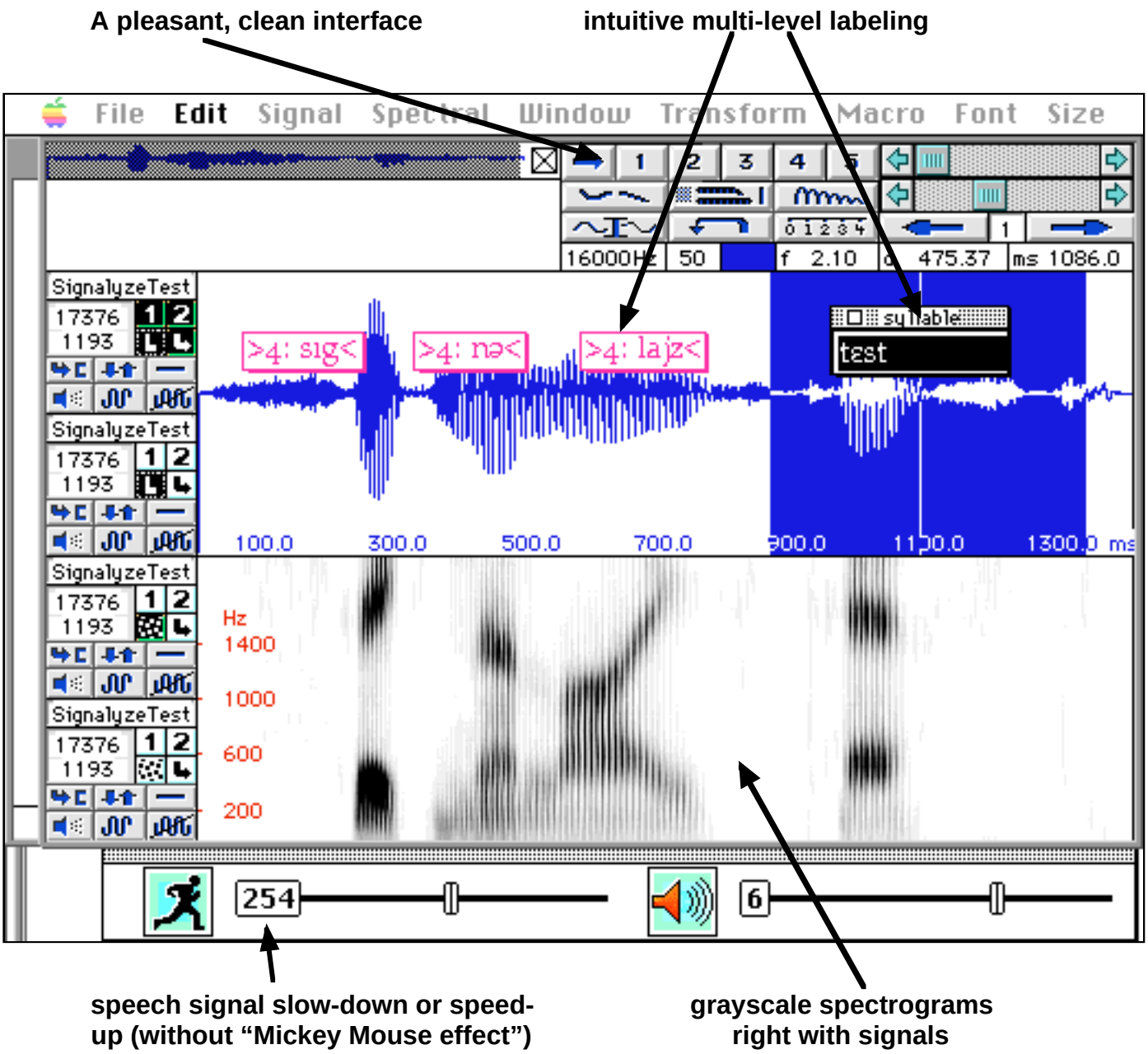
Eric Keller
Author, Signalyze™
Professor of Computer Science, University of Lausanne

---

1 Users of the new Centris computers running System 7.01, for example, found at the beginning of 1993 that Signalyze 2.0 gave an "echo" sound. Version 2.40 fixed the problem—but had the "regrettable side effect" of cutting off *all sound* for non-Centris users running System 7.1. Finally, Version 2.42 introduced a choice of channels that fixed the problem for everyone. Could this set of events have been foreseen in January 1992 when Signalyze 2.0 came out? Not likely. Upgrades are always required to keep up with technology.

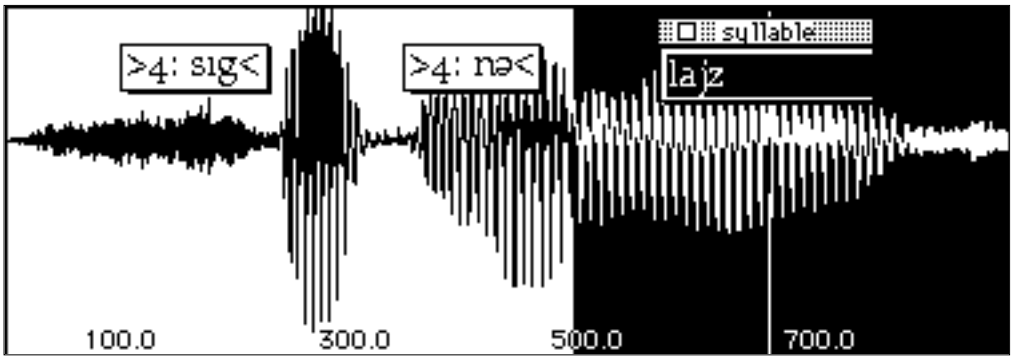
A Summary of Version 3.0 Additions

Here's Version 3.0:



Here's a list of new features of Version 3.0. Please refer to the new sections of the *Manual* for details.

Labeling (see Tutorial 4)



Labels at level 4 (here, the syllable level). Do CTRL-click to open any of these labels.

An open label, about to mark the blackened signal selection. Use CMD-Y to play the signal selection.

- The Signalyze labeling operation works by levels. For each label, you specify a level, such as "segment", "syllable", "phrase" etc. Each level has its own label color. Labels are marked for their level by a number placed in front of the label name (e.g., "4: its", means that the label "its" is marked for the fourth level).
- Signals with labels are designated by an "L" that appears in the Enable Button. You can turn the label display on or off by Edit>Show Labels.
- You can label either points in the signal or selected portions of the signal. To label a point, simply click at the desired place before opening the label. To label a segment, blacken the desired segment before opening the label. The labels for selected portions will be placed at the center of the selection and will be marked by angular brackets (">...<").
- Each label must have a different millisecond time-stamp. If you attempt to place a second label at exactly the same place as another label, you'll get an error message.
- While the label is open for editing, you can play the selected portion of the signal by doing COMMAND-Y. If you're labeling a selection, and if you've set the audio to play signal selections, you'll hear the segment which you're about to label.
- You can choose any of nine different vertical positions for placing labels. When you close a label with a click in the close rectangle or with RETURN, the label automatically "snaps" to the nearest standard position.
- The label information is saved as a TAB-delimited TEXT file with the extension ".lbl". It is stored in the same folder as the signal file. Refer to the Reference portion of this manual (menu Edit>Label Setup) for the detailed structure of label files.

- *Open File* operations on signal files that have accompanying ".lbl" TEXT files cause the label information to be read into Signalyze. *Save File* operations on signals with labeling information automatically saves the labels in the same folder as the signal. Save File operations for signals without labeling information erases whatever label file with the same name that may have existed in the folder. At the present, there is no method for saving labels elsewhere than with the signal.
- If you want to use a phonetic font for labeling, stick to *SigPalFont* (the shareware font supplied with Signalyze). SigPalFont preserves the numbers and angular brackets you need to indicate labeling levels, which is usually not possible with other phonetic fonts.
- You can search for labels using either a given labeling level or the label's name. You can specify two separate search patterns. Switch between the two patterns with the SHIFT-LOCK key.

New Menu Items

Edit>Search by Label: Specify search options to find a given label or all labels of a given level.
Edit>Find Label Again: Perform a label search according to specifications in Edit>Find Label.
Edit>Label Setup: Lets you define level names, level colors, label fonts and label font sizes.
Edit>Show Labels: For signals that have labels, turn the label display on or off.
File>Save Signal, Open Signal, etc.: For signals that have labels, open and save label files that accompany signal files.

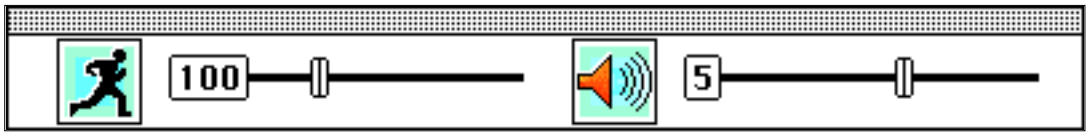
New Click Items

The label operations all use the distinctive CTRL key. This key is not used for any other Signalyze functions.

CTRL-click in signal: Make a label.
CTRL-click on label: Edit an existing label.
CTRL-SHIFT-click in signal: Determine the current labeling level.
CTRL-OPTION-click on label: Delete a label.
CTRL-OPTION-CMD-click on label: Delete all labels.

CLICK-and-drag at label edge: Enlarge or reduce label width.

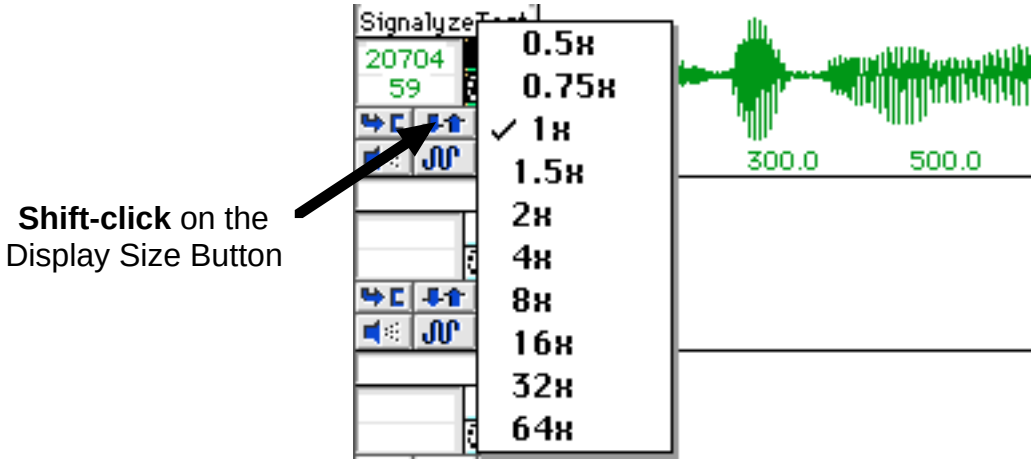
The Speed Palette



Click: Set speed to normal (100%)	**Speech Speed-up and Slow-down** 20%: Fastest 100%: Normal 500%: Slowest	**Click:** Play Sound according to Audio Setup specifications	**Loudspeaker Volume:** 0: silent 8: maximum

The speed-up/slow-down works smoothly on speech signals with reasonably clear fundamental frequency patterns between 60 and 500 Hz. This may include animal sounds. This feature is unsuitable for voices that have highly irregular fundamental frequency patterns. Better results are possible at the higher sampling rates (16 kHz and higher).

The Vertical Zoom Popup

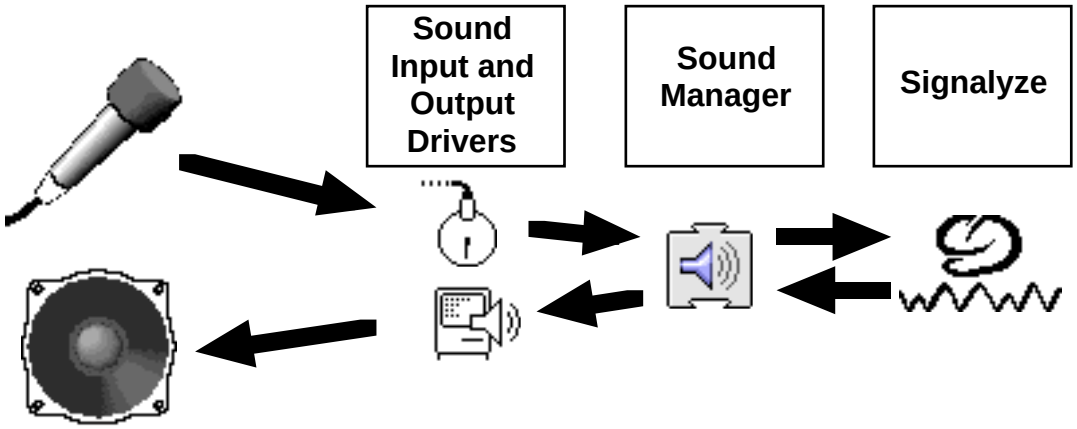


Choose vertical zooms between 0.5 and 64 times normal display size. At large zooms, you can inspect the detailed features of a signal's quiet portions, and at reduced zooms, you can display the signal without interference from the time labels.

Support for AV-Machines and Power Macintoshes

Signalyze 3.0 implements support for the new, 16-bit Apple Sound Manager. This lets you work with the 16-bit input and output features of the Macintosh AV Machines and PowerPCs. *This standard interface will over time become the only sound I/O that Signalyze supports.* Already,

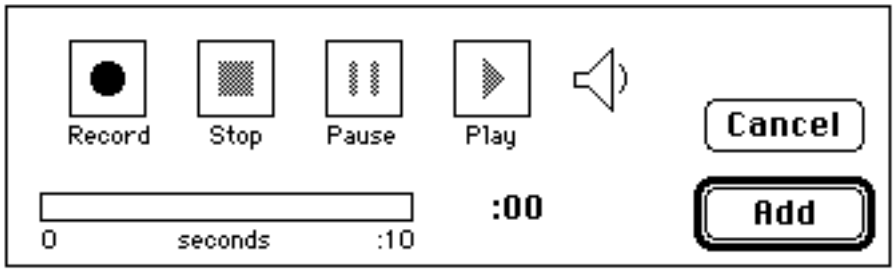
"drivers" (small utility programs) are available that interface the MacRecorder, AudioMedia and Spectral Innovations devices to the standard Apple sound interface. Please make sure that you get, directly from the manufacturer, the drivers for whatever other I/O devices you use. *Attention:* Signalyze Version 4.0 will no longer support manufacturer-specific hardware.



Working with the Apple Sound Manager is a four-step process. Input and output from specific devices is fed to and from the Sound Manager via specific *drivers* situated in the Extensions folder of the System Folder. The Sound Manager, in turn, provides a standard interface for in- and outputting sound to Signalyze (and other applications). The Sound Manager portion of the System may or may not be visible. On older systems, the Sound Manager Init is in the Extensions folder of the System Folder, and on newer systems it is part of the System file. You can set the Sound Manager options through the Sound control panel. Notices on the various drivers and input devices are found in the folder "I/O devices & info", found in the Signalyze Folder on the distribution diskette.

Selecting Drivers in the "Sound" Control Panel

Before attempting to record with the Sound Manager, verify the input connections. Open the "Sound" Control Panel and select the desired input device. Then return to "Alert Sounds", click on the "Add" button. You should see this dialog:



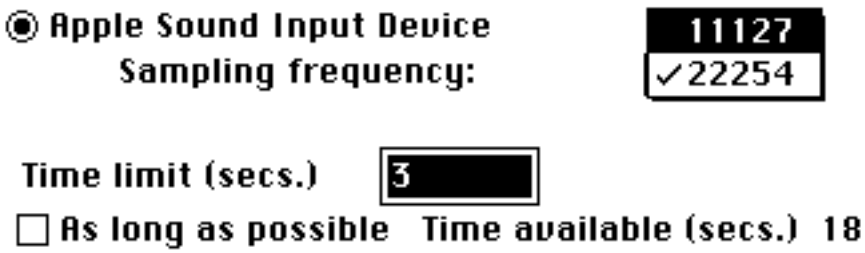
If you get "waves" coming out of the loudspeaker as you speak into the microphone or provide sound input, the entire setup for sound input works correctly. Once this is the case, Signalyze should usually not have any difficulties recording.

Also select the appropriate output driver. Whenever possible, use 16-bit options. For outputting, use the higher sampling frequencies (20 kHz and up). The inputting frequencies are selected from inside Signalyze.

Selecting the Recording Options in Signalyze

In Macro>Record Setup, select "Apple Sound Input Device" and the sampling frequency appropriate to your research project. Remember that it should be equal to or greater than twice the highest frequency you want to observe in the speech signal. (If you use a version of Sound Manager prior to 3.0, the only available recording frequency is 22254 Hz).

To find out if you have Sound Manager 3.0 in your machine, observe the small loudspeaker button during playback via the built-in speaker. If it inverts colors during playback (in color: if it turns yellow), you're running Sound Manager version 3.0 or higher, if not, a version prior to 3.0. Sound Manager 3.0 gives you much better sound quality via the built-in speaker than previous versions—even on older Macintoshes—and it permits 16-bit recording, providing you have the necessary hardware on your machine.



The recording process works as in Version 2.0. You should start speaking when you see the plus-sign cursor. Once you've started recording, you cannot stop in mid-course.

If Something Goes Wrong

If you can't get the input driver to work correctly (you don't see the "waves" coming out of the loudspeaker, or you can't hear anything at all), please contact the manufacturer in question (Apple for built-in microphones, MacroMind for MacRecorder, Digidesign for AudioMedia, Spectral Innovations, etc.). This type of problem has nothing to do with

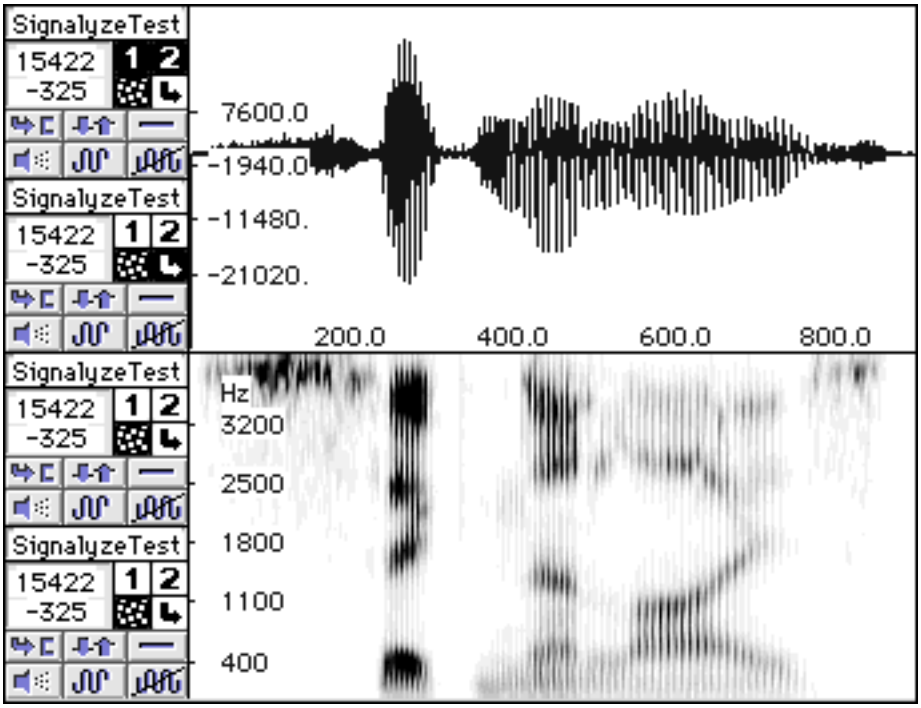
Signalyze. To obtain the latest version of the Sound Manager, please contact Apple.

Please note also that the new generation of Macintoshes (RISC-based machines and AV-Macintoshes without AV monitor) require *amplified* sound input. That is, you must record either from a microphone with a powerful built-in amplifier, or via the earphone output of an amplifier (careful: keep the volume control low at the start!). AV-Macintoshes with the special AV monitor also have a microphone-on switch at the bottom front panel of the monitor that must be activated before recording can take place.

If you get distorted sound, you are either providing too much power (reduce the volume on the amplifier), or you are using an badly matched type of input. Please consult your Apple dealer for input specifications that permit your audio dealer to select a correct and matched type of input amplification.

If you "see the waves", but still experience difficulties with in- and outputting, chances are that the problem is still not with Signalyze (at this writing, current versions of the AudioMedia driver, for example, do not work correctly on all machines via the Sound Manager 3.0). The new sound in- and output has been tested on a large number of machines with sound I/O drivers from four different manufacturers. Before contacting InfoSignal, please make sure you've made all the right selections in Macro>Sound Recording Setup and in Signal>Audio Setup.

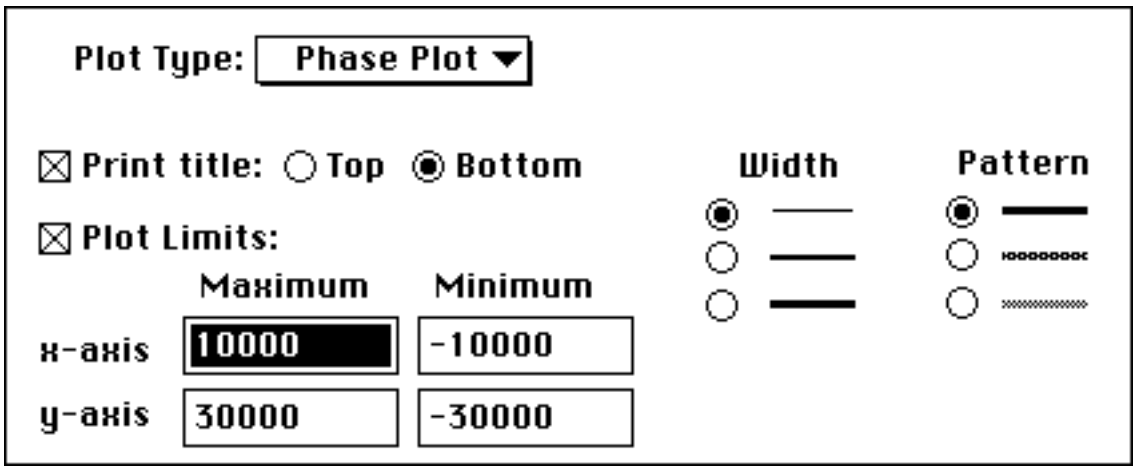
Color and Grayscale Spectrograms on the Signal Window



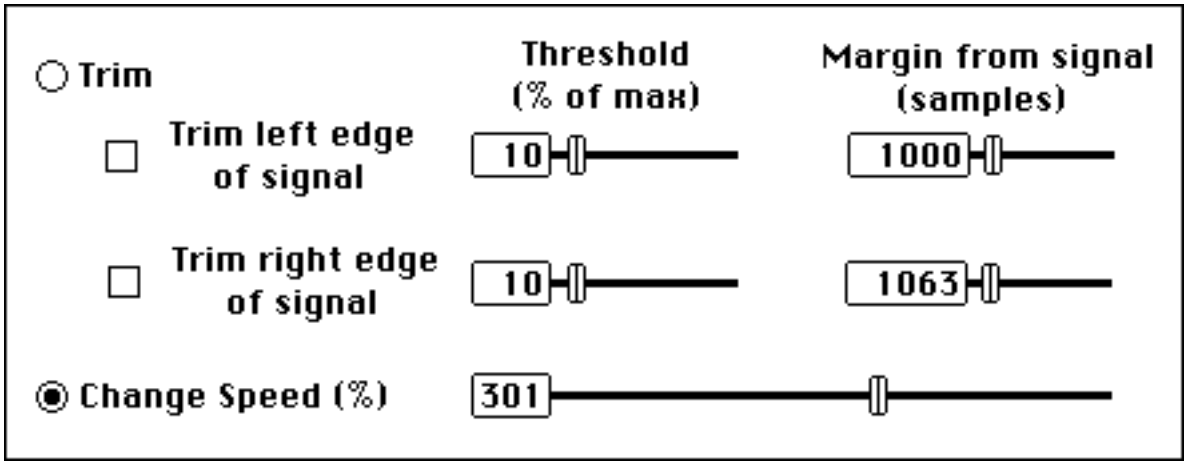
On machines and screens that support grayscale or colors, the Signal Window is now capable of grayscale/color spectrograms. Also, you can now switch directly between color and grayscale spectrograms in the Spectral>Spectral Analysis dialog, and you no longer have to go through the Monitors control panel. Furthermore, spectrograms are now preserved over card switches. However, watch out: 256-color spectrograms gobble up lots of memory, and your calculation memory can quickly become saturated if you keep many spectrograms around. To stretch your memory, switch to 16 color/grayscale levels.

Scale Limits for Big Plots

Some researchers have been hampered by the previous version's automatic sizing features for big plots (Spectral>Make Plot), since different graphs gave plots with different dimensions. Version 3.0 gives you an optional, manual control over the plot dimensions.



The Auto-Edit Transformations



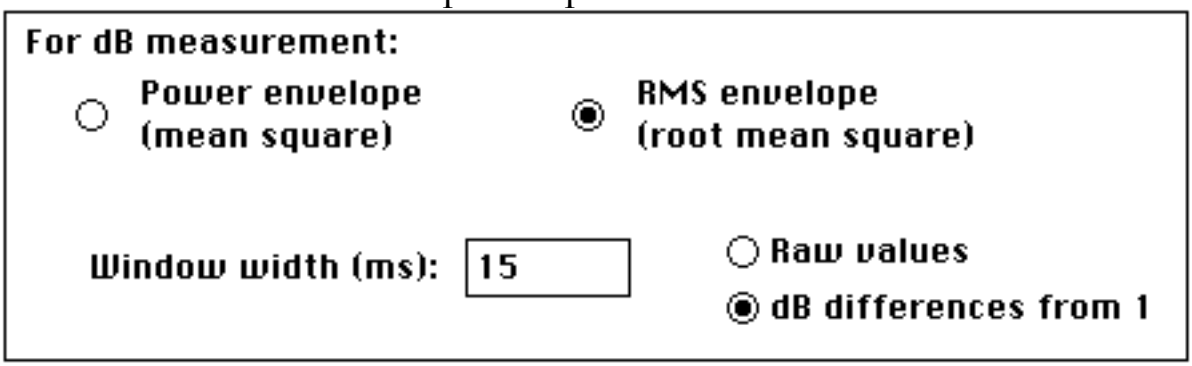
When you have a signals with quiet edges, it saves a lot of disk space to be able to trim off the edges. Also, you may want to use the Signalyze speed-up and slow-down routines to make new, speed-modified signals. The routines in the new Auto-Edit transformations menu will let you perform these functions.

Decibel Differences to the Value 1

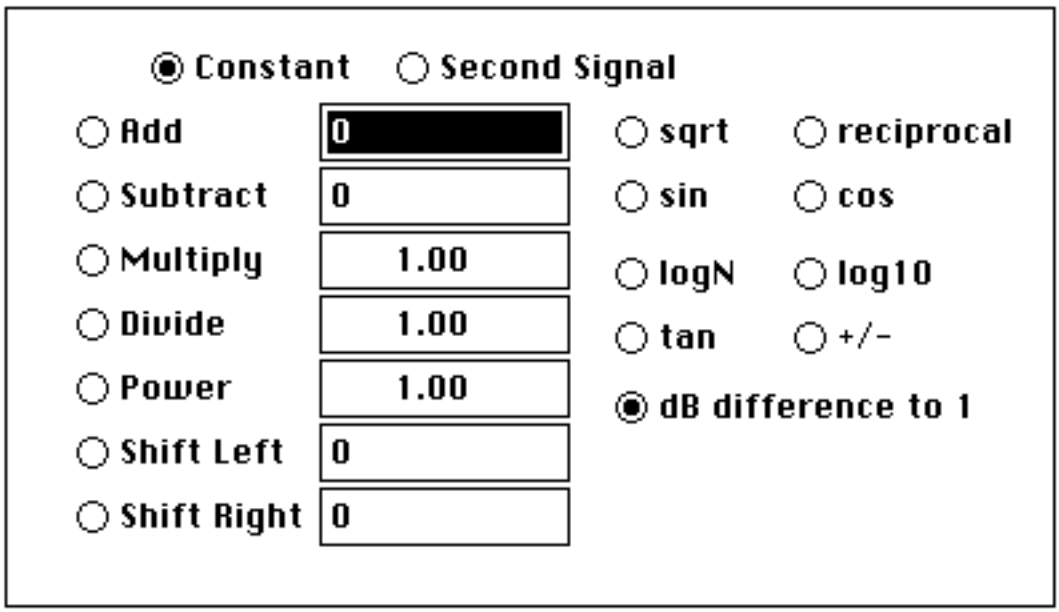
Amplitudes should generally be interpreted as relative to other amplitudes, and they are related to bit width (larger bit widths can encode larger amplitude differences). For these reasons, previous versions of Signalyze have been reticent about providing an easily misinterpreted decibel scale for amplitude values.

However, Version 3.0 now bows to public pressure and provides decibel differences between observed values and the minimum value of 1. You can now obtain RMS. or power envelopes as decibel-denominated curves and you can convert any signal into a decibel curve by means of a new simple transformation.

In Transform>Envelope Setup:

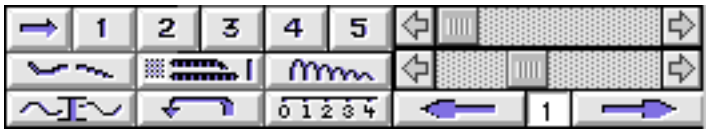


And in Transform>Simple Transformation Setup:



The Snazzy New Look

And for those with black-and-white monitors, here's what the buttons now look like on a grayscale or color monitor. Try it—you may like it.



And as always...

Signalyze remains compatible with the large majority of Macintosh developments. For example, users of RAM Doubler found out earlier this year that without any changes, Signalyze was compatible with this very efficient means of extending the available RAM...

First Things First: What is Signal Analysis?

In its basic definition, a **signal** describes **any detectable physical quantity that changes over time.**

Sound is a signal. An EEG trace is a signal. A stream of computer bits coming in over a wire is a signal. Continuous measurements of arm or lip movements are signals. Any analysis involving continuous measurements of a property in change is thus *signal analysis*.

We humans perform signal analysis all the time. As we live and move, we analyze signals from a vast multitude of light sources to find our way, we sift through acoustic signals from speech, environmental sounds and music, we sort through tactile signals from the objects we touch, and we record internal signals from our own bodies. Even as we sleep, we continue to monitor various bodily functions and remain aware, to a certain degree, of sounds surrounding us. In a sense, our very survival depends on a tremendously complex signal analysis performed in parallel on a continuous stream of input signals.

By means of signal analysis performed on computers, we can begin to understand our universe in its most fundamental expression. We can try to capture the signals that the human body responds to or generates, and we can puzzle out how we hear, see, feel or function. We can also go much beyond the signals that humans can sense, and capture information streams from physical objects, such as meteorological and seismological observations, or signals generated by any suitable source, such as radio waves, ultrasounds, or X-rays.

One of the fundamental aspects of signals is that they are multidimensional. Although sound is typically thought of as a single wave traveling from a sound source to the human ear, it is multidimensional in at least two ways. First, it's precisely **not** a single wave traveling to a single ear. It's a great number of waves coming from multiple directions and traveling, among other places, to **two** ears. And even if this multiplicity of waves is forced into a single microphone, and from there to a listener's single ear, the acoustic signal encodes in complex ways the sound's various sources.

The game for the listener is to find out what's hiding behind the sound. In speech, that's called "perceiving the sounds" and "understanding the meaning". In the science of signal analysis, finding out what's coded *inside* a signal is the main game in town.

What Signalyze Can Do for You

This program gives you some fundamental tools to do signal analysis (Signal-yze -- "**signal** anal**yze**"). It is mostly concerned with signal analysis in the acoustic domain, but there is no inherent reason why you couldn't use the program for the analysis of EEG, EMG or motion signals. In fact, some of the features of the program, such as the splines, are there exactly for the analysis of articulatory movements in speech.

Still, most of the program was developed to help you do analysis of acoustic aspects of speech, music or animal sounds. Specifically, the program does **multidimensional analysis of the sound waveform**, since it works not on just one or two channels, but handles up to **100 concurrent channels** (and that's a soft limit, if a need for more channels ever arose, this limit could be extended). Typically, in the course of working with signals, speech researchers derive a certain number of **secondary signals** from the original waveform. They "extract" the signal's fundamental frequency, or pitch, its amplitude envelope, its frication index, its spectral characteristics, and each of these secondary signals is examined for particular features.

Signalyze does all that, and it does it about as painlessly as is possible on a small computer. There are many commands to **move around** the various signals, to **zoom** in on their detailed features, to **edit** signals, and to **play** acoustic signals. Signalyze is also a tremendous help in **manual scoring**. If you have to measure specific features by hand, Signalyze will be much easier to use than many any other programs.

Much can be learned about speech and the functioning of articulatory organs, or about music and the characteristics of individual instruments, by examining the **spectral characteristics** of signals. The program thus contains a fairly extensive set of spectral analysis tools, including **spectrograms** with wide band, narrow band and cepstral options, **spectral sections**, and three **pitch extraction** routines.

As required by contemporary acoustic signal analysis, the program furnishes **amplitude envelopes**, **zero crossings**, and an array of signal manipulation tools, like **up-** and **downsampling**, **derivatives**, and **splines**. You can also transform your signal or combine it with a second signal through a variety of **transformations**.

No matter which application you have in mind, ease of use and clarity in structure is of foremost importance. As you get acquainted with the program, you'll probably find it quite easy to use. In time, the program will grow into a tool similar to a word processor or a spreadsheet. You won't think about **how** to use it any more, but rather, about what the program can **do** for you.

To get the most out of your time, do the following:

- **Above all, do the tutorials**. They won't take very long. If you don't do them, you'll end up wasting a lot of time doing things the hard way. Besides, you'll never know some of the nifty things that Signalyze can do.
- Get familiar with the **index** and how to find things in the **reference section**. Many more hints and explanations are provided there. It's all clear and simple if you have the manual open at the same time as you work with the program.
- **Use the Help facility**. A special effort was made to put all answers to common question into the Help. Explore, print out, mull over the explanations, it's all there for you.
- There isn't much that isn't answered in the Manual or the Help. But if you do have a question, you can always **email, fax** or **write to me** at the addresses given on the title page. Please don't phone. My days are chock full with the usual duties of a professor and researcher. But at night and on weekends, I catch up with my mail. Usual turnaround is 24 hours, sometimes it's just a few hours.

Note to those working in fields other than speech

The program was developed primarily with speech analysis in mind. However, most of its features are also of interest to people working with animal sounds and with music, and many features are applicable to any type of signal analysis. InfoSignal is gathering suggestions of features that will make Signalyze even more useful in the analysis of animal sounds and music. Please do not hesitate to let us know what you need to make Signalyze work for you.

Welcome to Signalyze

The fastest way to get to know Signalyze is to come along on a guided tour. If you're totally new to speech signal analysis, working through these examples will open a whole new world for you—but schedule about an afternoon for each tutorial. If you're already familiar with speech signal analysis, working through the tutorials will only take a few sessions. In either case, the tutorials are **strongly recommended reading** because they show you most of the program's features.

If you absolutely can't spare the time, just glance over the next few pages. They list the terms used throughout the Help and the Manual. Also, find "**Summary of Key-Click Combinations**" in the Appendix. After a while, you won't be able to live without them.

Avis pour les utilisateurs de langue française

Signalyze permet de travailler en langue française. Il suffit de sélectionner «Français» dans «Langue» dans le menu «Edition». Quand les menus apparaissent en français, le logiciel fonctionne automatiquement avec le fichier «Aide-Signalyze»—dans la mesure qu'il se trouve dans le même dossier que le logiciel. Une version française de ce manuel est également disponible de la compagnie InfoSignal.

Anmerkung für deutschsprachige Benutzer

Signalyze erlaubt es, in Deutsch zu arbeiten. Wählen Sie einfach "Deutsch" in der Auswahl "Sprache" ("Language") des Menüs "Bearbeiten". Wenn die Menüs auf Deutsch erscheinen, funktionniert Signalyze automatisch mit der Datei "Signalyze-Hilfe"—solang diese Datei im gleichen Ordner erscheint wie das Signalyze Programm. In Anbetracht des ausserordentlich beschränkten deutschsprachigen Marktes und der schnellen Entwicklung des Signalyze Programms, sowohl wie der Macintosh Technologie, liegt es leider ausserhalb unserer Möglichkeiten, das Handbuch auf Deutsch zu übersetzen.

A Guided Tour of Signalyze

The License Agreement

Decide if you want to be bound by the license agreement (it's toward the end of the manual). Notice the money-back guarantee for substantial problems, but please also note that you are responsible for verifying your own results. Feel free to leaf through the manual, and be sure to read the **About Signalyze** section. If you don't like what you see, pack it all up and send it back to where you got it.

If you're positive you want to jump aboard, copy the distribution diskette to your hard disk, or to another diskette. Make sure the Signalyze program is in the same folder as Signalyze Help. When you're done copying, store your distribution diskette in a safe place.

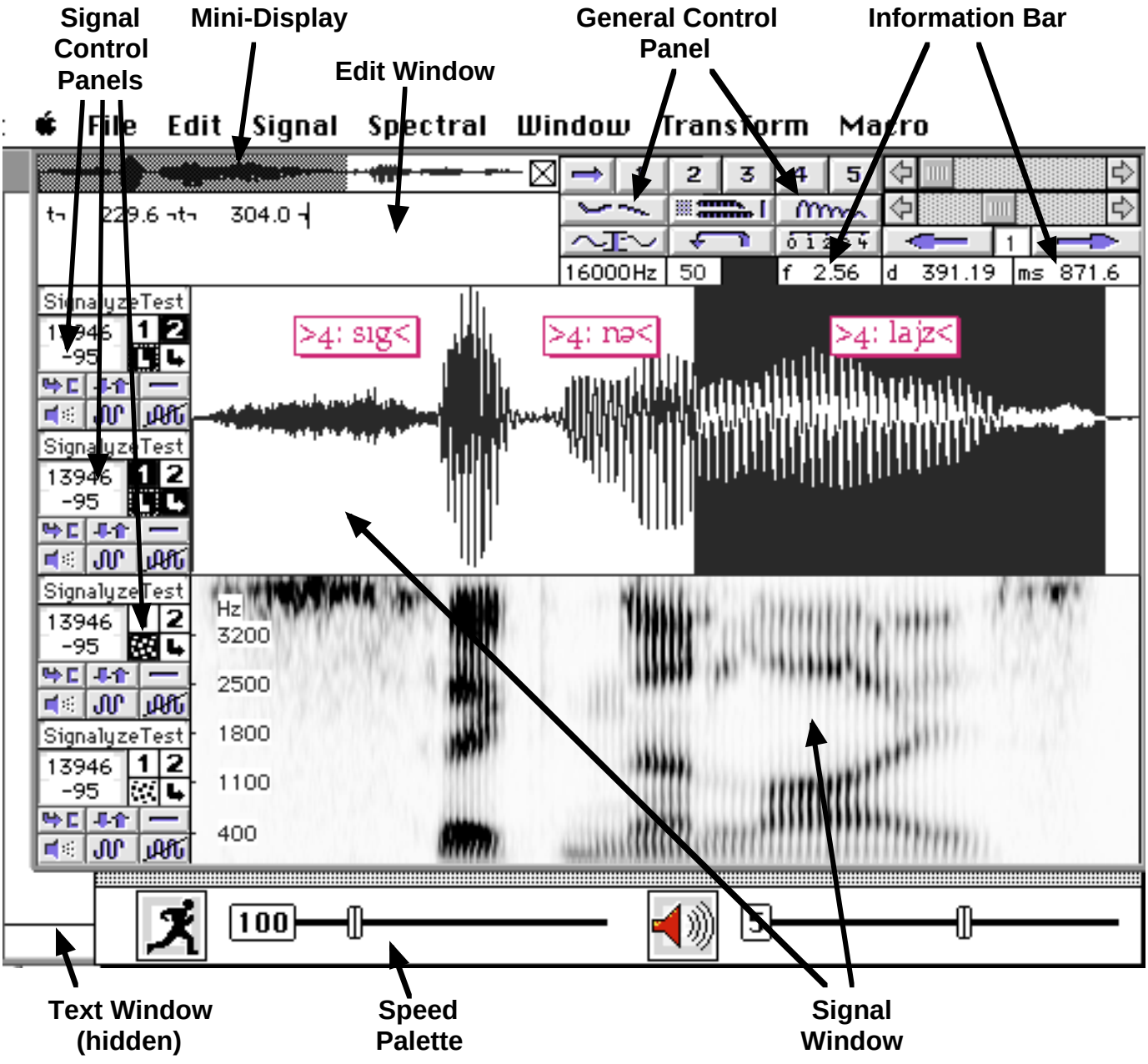
or Signalyze Application icons

Start the program by double-clicking on the Signalyze application icon. If you encounter a problem at this point, please consult the section on **Installation Suggestions**.

Tutorial 1: Signalyze Basics

This chapter gives an overview of the various windows and buttons, as well as of their main functions.

Terms and Windows



The Signal Window

1. The **General Control Panel**. The actions controlled with this panel apply to **all** signals.
2. The **Information Bar** shows a number of values concerning the currently selected signal and the line cursor. Control over signal track color and spectrogram darkness is also found here.
3. The **Signal Control Panel** to the left of each signal track contains four smaller and six larger buttons. The small white area shows information about the signal. The actions controlled with this panel only apply to the **signal shown to its right**.
4. **Edit Window**. At various times, information about ongoing processes is given in this window. It also shows numeric values you've selected from the signal.
5. The **Mini-Display** gives a small and complete representation of the currently selected signal. The checkbox en/disables the Mini-Display.

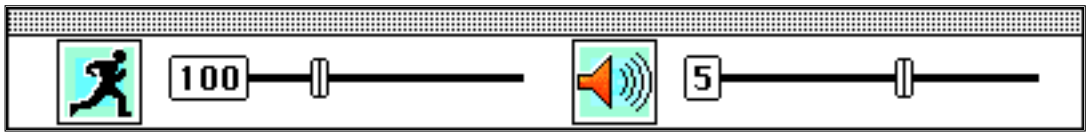
The Signal Window has a minimum of four signal tracks. Its initial position on the screen is specified numerically in Edit>Startup Options. Open and save signals with File>Open Signal and File>Save Signal. Signal files can be saved in a number of different formats.

The Text Window

The menu command Window>Text Window (or clicking in the Text Window) activates the **Text Window**. This editing space cumulates numeric information from the signals. The Text Window can be moved anywhere on the screen and will stay there till the next session. Open and save text to and from this window with the commands File>Open Text (visible when the Text Window is in front) and File>Save Text (visible with both the Text and the Signal Window). The open and save files are of type "TEXT".

The Speed Palette

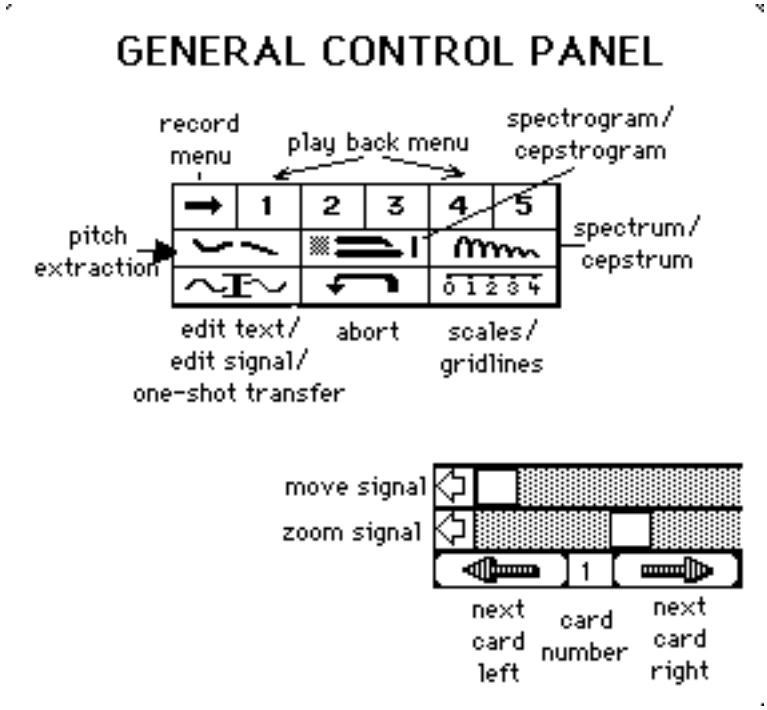
You can modify the sound reproduction speed of normal speech with the left control of the speed palette (see illustration next page). Also, you can set your loudspeaker volume and initiate sound playback on this palette.



Click: Set speed to normal (100%)	**Speech Speed-up and Slow-down** 20%: Fastest 100%: Normal 500%: Slowest	**Click:** Play Sound according to Audio Setup specifications	**Loudspeaker Volume:** 0: silent 8: maximum

The Control Panels and the Information Bar in the Signal Window

The exact meaning of the various buttons is explained in later tutorials, and a full description is found in the Reference section. For now, just glance over these illustrations to get the general idea.



(This illustration is permanently available in "Help" under the Apple menu.)

- **First row**

— Record a menu-command.

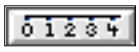
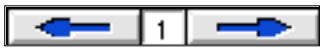
— 1 to 5 Play back recorded menu-commands.

- **Second row**

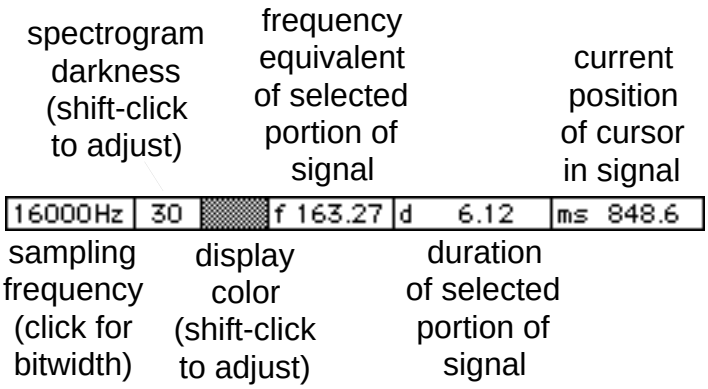
— Pitch extraction.

— Spectrogram generation.

— Spectrum display.
SHIFT-click calls corresponding setup dialogs.

- **Third row**:
  — / Switch between text and signal editing.
  — Abort ("U-turn") button.
  —  Control scales and gridlines.
  —  Switch between cards.

The Information Bar

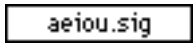
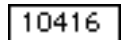
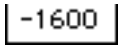


The various fields of the **Information Bar**:

- The current signal's sampling frequency (shift-click to modify)
- The current spectrogram darkness
- The current signal's display color (on color/grayscale machines)
- The frequency equivalent of the duration of the selected portion of the signal ("f", =1000/ms, 1/s, etc.)
- The duration of the selected (blackened) portion of the signal ("d", ms, s, μs, etc.)
- The current cursor position in ms, s, μs, etc.

The Signal Control Panel

The **Signal Control Panel** has three areas:

- **Information area:**
  —  Signal Name (shift-click to modify)
  —  Sample at cursor
  —  Value of sample at cursor (shift-click to modify)
- **Upper button area**:
  — Indicates the **selected** (or current) signal.
  — Indicates the **second** signal in operations involving two signals.

— / Indicates whether a display is to be **refreshed** when the zoom or the selection changes**.**

— / is the same as / , except for a signal with labels.

— Points to the track where the next signal is **inserted**

- **Lower button area**:

— Copies a signal from another signal track

— Toggles the signal display between small (64 pixels) and tall (128 pixels)

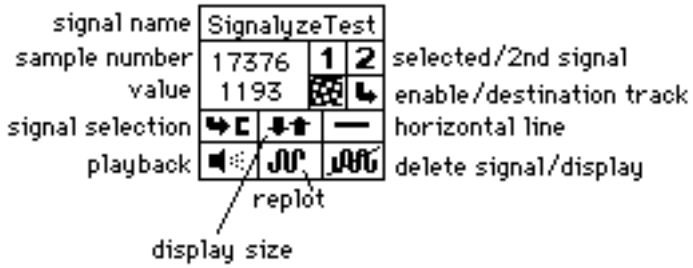
— Toggles the horizontal cursor

— Initiates sound reproduction

— Replots the signal

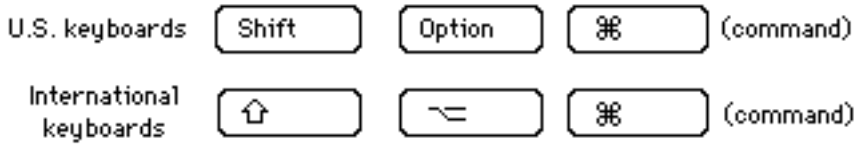
— Deletes the signal from memory; with ⌘, it deletes the current *copy* of the signal from the card (you can have several copies of a single signal)

SIGNAL CONTROL PANEL



(This illustration is permanently available in "Help" under the Apple menu.)

For Future Reference: Modifier Keys on International Keyboards



The modifier keys look different on U.S. and international keyboards. In this text, the *shift* key is the -key on international keyboards, the

option key corresponds to the ⌥-key, and the *command* key is the ⌘-key.

Tutorial 2: First Steps

This chapter shows how to move around the signal: zooming, selecting signal segments, and moving the signal to the left or the right.

Loading a File

— Click in an empty signal track
— Open a file with the File>Open menu (or COMMAND-O, or SHIFT-).
— Select the file "Signalyze Test" (distributed with the program).
— The file is loaded into the track where you clicked.

Note:
— To see long signal names, shift-click on the name.
— The first number below the name gives the sample *number* (i.e., the how-manieth sample that the cursor is sitting on). This is often useful to check results.
— The second number gives the *value* at the cursor (SHIFT-click to modify).

Selecting a Signal Track

Method 1: Click in signal.
Method 2: Click on the number-1 button ( ).
Method 3: Press the up-/down-arrow keys on the keyboard.

Zooming

There are three ways to zoom the signal:

- **Zooming with Menu Commands**

— Drag-select a signal segment
— Activate the menu command Signal>Show Selected Segment
— The program zooms into the selected signal segment
— The mini-display (upper left hand corner) shows which part of the signal has been zoomed.
— Select the menu command Signal>Show Whole Signal
— The program shows the whole signal again.

- **Zooming with Scroll Bar**

— Move the lower of the two scrollbar controls at the upper right.
— Moving the control to the right "stretches the signal away from its left margin".
— Moving the control to the left "squeezes the signal into its left margin".

- **Zooming with Key Clicks**

Zoom In:
— Drag-select a signal segment.
— Hold down both OPTION and COMMAND keys.
— Click on in the Signal Control Panel.
— The selected segment is zoomed.
Zoom Out:
— Hold down both SHIFT and COMMAND keys.
— Click on .
— The whole signal is shown.

> **Remembering Keyboard-Click Combinations:** For a complete list of keyboard-click combinations, please refer to the table in the Appendix.

> **Zooming imprecisions**: When zooming out from *greatly amplified zooms* via the Mini Display, there may be some imprecisions in the calculation of the new display. This is because the cursor's mathematically coarse pixel position feeds into a power function and causes an imprecision in the calculation. Simply click once more to get the display you want, or play a bit with the zoom scroll bar.

Selecting Portions of the Signal

Selecting a portion of the signal works just like selecting text with a text processing system:

Mouse Dragging

- Click anywhere in the signal, keep the mouse button pressed and drag the mouse to the right or left.
- The "dragged" portion of the signal begins to turn black (or in inverted color) and remains black when you release the mouse.

- The blackened segment of the signal is considered to be its "selected" portion.
- Click anywhere in the signal to get rid of the blackening.

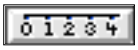
Shift-clicking
- Click at the beginning of a portion of the signal.
- *Shift-clicking* at the end of the signal segment (i.e., holding the shift key while clicking).
- While holding *Shift* down, drag the cursor left or right for final adjustments.
- The "in-between" portion of the signal turns black.

If There's No Blackening
If there is no blackened segment on the screen, the *visible* portion of the signal is considered to be the "selected" portion.

Operations Apply to Selected Signal Segments
Operations apply to previously selected material. E.g., transformations, spectrograms, edit functions, etc., apply just to the selected segment.

Shift is the Key
to easy functioning in Signalyze.

SHIFT gives you access to edit boxes, popup menus and common dialog boxes. Try holding SHIFT while clicking on these buttons:

BUTTON	NAME	WITH SHIFT
	The Sound Playback button	Adjust volume
	The Display Size button	Vertical zoom
	The Signal Selection button	Load signal
	The Pitch Extraction button	Pitch setup
	The Spectrogram button	Spectral setup
	The Spectrum button	Spectral setup
	The Scales button	Scale and grid setup
°	The signal name	Change name
°	The spectrogram darkness indication	Change darkness
°	The color indication	Change color

Duration
The program maintains a *duration indicator* in the Information Bar (marked "d"). It shows the selected segment's length in time (usually in

ms). The time unit (e.g., milliseconds, seconds) can be changed in Edit>Numeric Formats.

Frequency Equivalent for Duration
The *frequency equivalent* of the duration of the selected portion of the signal is also given in the Information Bar (marked "f"). This is often useful for verifying the fundamental frequency in the raw signal (blacken the space between two major peaks in a vocalic portion of the signal at a large zoom setting).

Moving Left and Rightin the Signal

Move with Scrollbar
- Move the top scrollbar to the right: The *view* moves right (the signal moves left)
- Move the top scrollbar to the left: The *view* moves left (the signal moves right)

Move with Key-clicks
- OPTION-COMMAND-click anywhere in signal or in Mini-Display: The *left edge of the view* moves to where you clicked.

Different Types of Displays

There is a standard way of displaying signals on the screen, called the "MinMax" display. It consists of seeking the minimum and maximum values among the samples "lying under" the vertical pixel line, and of drawing a vertical line connecting the two. At low zoom settings, signals appear as "sausages", and at high zoom settings, they appear as lines that connect points.

Signal appearance can be changed with Signal>Display Setup. There, it can be seen that the standard signal format is

"Vertical/ MinMax/ Every Pixel".

Pitch extractions use a different format:

"Connected/ Sampled/ Every Pixel"
or "Dots/ Sampled/ Every Pixel".

After loading a pitch line previously saved to disk, select its signal track and set the desired options in Signal>Display Setup.

Tutorial 3: Sound Reproduction and Sound Input

(See also the section “Support for AV-Machines and Power Macintoshes”, p. 10)

This chapter shows how to reproduce and record sound. The different internal and external routines for out- and inputting sound are explained.

Sound Reproduction

Now, let’s hear the “Signalyze Test” signal. Press on the button in the Signal Control Panel or on the loudspeaker button of the speed palette. You should hear the sound of the signal.

If you Don’t Hear any Sound

— The loudspeaker volume may be too low. Shift-click on the button and adjust the volume.
— Something may be plugged into the earphone plug at the rear of the computer.
— The audio may be channeled via an alternative audio output. Check Signal>Audio Setup to see if that’s the case. If necessary, refer to the documentation on the alternative output to find out why there’s no sound.

Three Ways to Produce Sound

You can listen to...
— The *whole signal*, even if part of it is not visible
— The *selected (blackened) segment* of the signal
— *All but the selected* (blackened) segment of the signal
Switch between these three options in Signal>Audio Setup. Alternatively, use the following key-click combinations:

OPTION- : Play the whole signal

COMMAND- Play the selected segment

OPTION-COMMAND- Play all but the selected segment

Auto-Repeat Sound Reproduction

In Signal>Audio Setup, specify the number of repetitions.

Stop Playback in Mid-Course
— COMMAND-period
— COMMAND-click
— Click on -button. This does not work with MacADIOS™ synthesizers. With the Macintosh routines it may take some time, since aborts are checked only every 30000 samples.

Accelerate or Slow Down Speech
You can accelerate or slow down speech by playing with the left slide control on the Speed Palette. Click on the button with the "runner" to re-establish normal speed. Slowed-down reproduction does not sound as clean as normal reproduction—this is unfortunately inevitable.

Set up Sound Output Devices
• Audio Output via Macintosh: Switch to appropriate output ("Old" or "System 6+" in Signal>Audio Setup.
• Audio Output via an *AudioMedia* (or other *Digidesign*) board
— Switch to "Digidesign synthesizer" in Signal>Audio Setup.
• Audio Output via a *MacADIOS™ board*.
— Set the slot number in Macro>Record Setup.
— Switch to the appropriate MacADIOS™ synthesizer in Signal>Audio Setup.
• Audio Output via the *MacADIOS™ Model 411*.
— Turn both computer and Model 411 off. Turn the Model 411 back on and then the computer.
— Switch to the appropriate option in Signal>Audio Setup.

How do I stop this thing?

Nearly all Signalyze operations that take more than an instant can be stopped (aborted). There are three ways of doing this:

COMMAND-period
COMMAND-mouse click
click on Abort Button ( )

Some processes take quite long to stop or can't be stopped at all.

If it takes long to stop, it is either due to technical reasons (e.g., impossible to interrupt buffering as in Macintosh audio) or in order to speed up the process (e.g., spectrogram generation).

If it can't be stopped at all (e.g., record operations or audio via some boards), it is likely so for some technical reasons.

Sound Input

- **With Macintosh Microphone, 8-bit systems**

— Make sure you have a Macintosh capable of sound inputting (e.g., Mac LC, Mac SI, etc.).
— Connect the microphone to the Mac.
— In the "Sound" control panel, verify that the microphone works.
— In Signalyze, open "Macro>Record Setup...".
— Select "Apple Sound Input Device".
— Set time to record.
— Exit with "Record" from the dialog.
— Keep the mike well away from the mouth.
— Start speaking when the cross cursor appears in the record dialog.
(Careful: The Apple 8-bit audio input is of limited quality).

- **With Macintosh Sound Input, 16-bit systems**

Consult full instructions at the beginning of the Manual, starting p. 10, or the following summary ("System 7-Compatible Device").

- **With Any Other System 7-Compatible Device**

— Make sure of the following:
You're running System 6.07 or later.
You have a recording device for which a System 7-compatible driver exists.
This driver must be in the System folder and must be available to the "Sound" Control Panel.
— Connect the device to the Mac.
— In the "Sound" control panel, verify that the microphone works.
— In Signalyze, open "Macro>Record Setup...".
— Select "Apple Sound Input Device".
— Select the sampling frequency.
— Set time to record.
— Exit with "Record" from the dialog.
— Start speaking when the cross cursor appears in the record dialog.

- **With MacRecorder® or Similar Device**

— Plug device into the modem or the printer port on your Macintosh.
— In Signalyze, open "Macro>Record Setup...".
— Select "MacRecorder®...or Similar Device".
— Set sampling frequency (typically 11 or 22 kHz).
— Select the correct recording port.
— Set time to record.
— Exit with "Record" from the dialog.

— Start speaking when the cross cursor appears in the record dialog.
— If the sound is too weak or too strong, adjust the volume control on the recording device.

- **With AudioMedia or Sound Accelerator Board**

— Make sure device is installed in accordance with Digidesign's specifications. Make sure you have the Digidesign sound extensions in your System Folder ("DigiSystem INIT" and "Digidesign Sound Input").
— In the AudioMedia program, select recording via 44.1 kHz.
— In Signalyze, open "Macro>Record Setup...".
— Select "Digidesign"; if AudioMedia, set input volume.
— Set sampling frequency to any frequency between 5 and 44.1 kHz.
— Set time to record.
— Exit with "Record" from the dialog.
— Start speaking when the cross cursor appears in the record dialog.

- **With MacAdios™ Model 411 Device**

— Plug device into the modem port on your Macintosh (N.B., this device does not work with all types of Macintosh).
— Turn device on.
— In Signalyze, open "Macro>Record Setup...".
— Select "MacAdios™" and "Model 411".
— Set sampling frequency (typically 10 or 20 kHz).
— Set time to record.
— Exit with "Record" from the dialog.
— Start speaking when the cross cursor appears in the record dialog.
— If there no recording occurs, try the following: Exit from Signalyze and perform a complete shutdown. Turn off the Model 411 input device. Wait one second. Turn the 411 back on. Then turn on the Macintosh and start over again.

- **With MacAdios™ A/D Board**

— Make sure device is installed in accordance with GW Instruments' specifications.
— In Signalyze, open "Macro>Record Setup...".
— Select "MacAdios™", the appropriate board, and the slot in which the board is seated.
— Set sampling frequency (typically 10 or 20 kHz).
— Set the gain. If you're using a GWI-supplied pre-amplifier, gain setting 1 is probably appropriate. If the recording seems very weak, try gain 10 or gain 100.
— Set time to record.
— Exit with "Record" from the dialog.
— Start speaking when the cross cursor appears in the record dialog.

(Some of the newer 16-bit boards do not record correctly in Signalyze.)

Backgrounding

Macintosh Sound

Each *Signalyze* signal sample is 16 bits long. On the other hand, *Apple* sometimes uses 8-bit long samples to represent sound on the Macintosh. This gives sound that's reasonable, but not extraordinary.

Within the 8-bit standard, Apple has three traditions for sound reproduction. First, there is the *Sound Driver*. This is nearly-outdated technology that goes back to the original Macintosh. It is still used by Signalyze for the Motorola-68000 processor Macs (MacPlus, SE, Classic, Portable). These machines are too slow to convert the Signalyze 16-bit samples continuously into Apple's 8-bit output format via any other technology. The disadvantage of the sound driver is that it puts out a small audible click at the onset of the sound and with longer signals, there are barely audible clicks in the middle of the playback.

These clicks are unfortunate, but normal. To speed up operations, Signalyze does not keep the synthesizer going during normal operations (as do some music programs), but turns it on before each playback. Also, longer signals are played by switching back and forth between two buffers that store the 8-bit equivalents of the program's 16-bit signals.

A second, more recent 8-bit standard is known as the *original Sound Manager,* available since System 6. On computers with the faster processors running more recent systems, Signalyze switches automatically to this standard. Its sound is of higher quality and the clicks have disappeared.

A third, improved, 8- and 16-bit standard is known as *Sound Manager 3.0.* Signalyze switches to this I/O standard, if it senses a Sound Manager 3.0 or higher on-board. Please refer to the beginning of the Manual for details on this standard (p. 10).

To get improved sound out of a 8-bit synthesizer, connect the computer to an amplifier (aux input) and a good loudspeaker. Before you make the connection, lower the Mac's playback volume to its minimum through the Control Panel accessory, otherwise you may damage your amplifier. For safest operation, use an attenuator cable available from your local electronics store. Another good way to get better reproduction is to use a set of good earphones.

Any Other I/O Devices?

Users and potential buyers often ask if Signalyze will support any other I/O devices in the future. Here's the standard answer:

No, there are no plans to build in support for any other I/O devices in the foreseeable future. It is the I/O device vendor's responsibility to come up with a System 7-compatible sound input driver. Signalyze is primarily an analysis program and cannot be burdened down with a multitude of I/O drivers. Furthermore, there are no plans at InfoSignal to buy any additional I/O boards, since there exists totally satisfactory access to AudioMedia, Sound Accelerator, Spectral Innovations and MacAdios I/O devices at this point.

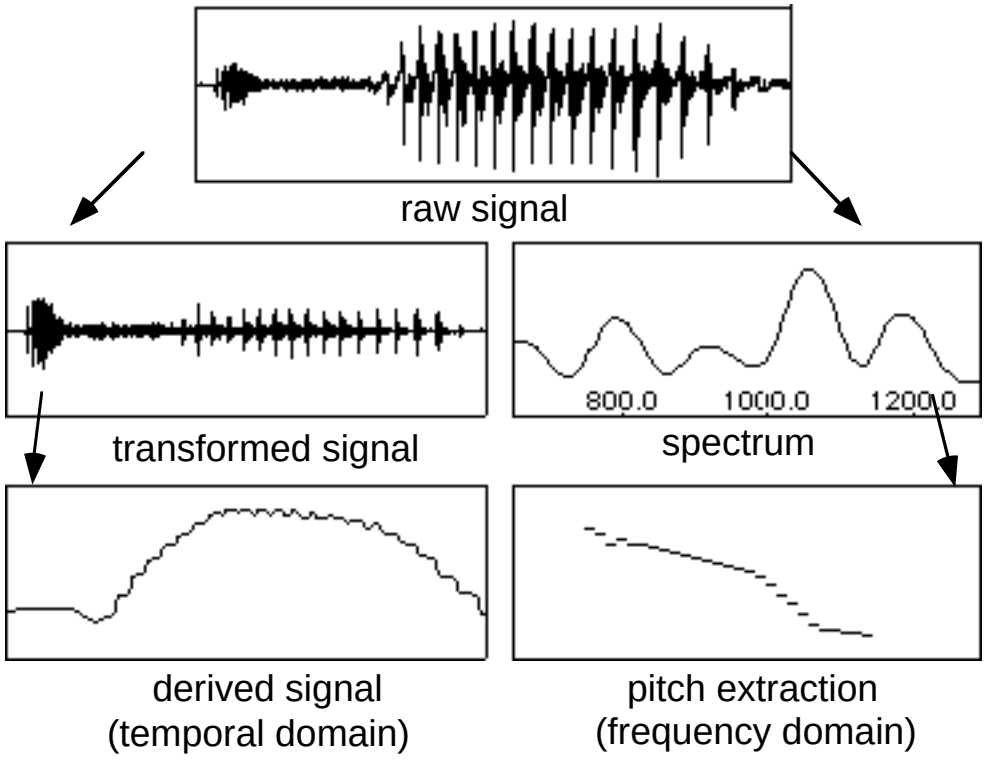
At the same time, InfoSignal is happy to cooperate in testing the Signalyze interface with any vendor-produced System 7-type input drivers, so as to permit Signalyze to work with other I/O devices.

Tutorial 4: Manual Scoring

"Manual scoring" means obtaining numeric information about a signal and its features by taking measures at manually identified places in the raw signal or in its derivative signals. This chapter introduces most of the tools needed to perform manual scoring: transforming the graphic display into numeric information, choosing form and type of numeric information, editing it, and saving it to disk.

General Information

The basic process of signal analysis is as follows:



- *Temporal domain:* Features in the temporal domain are either evident in the raw signal or they become (more) evident by deriving a second signal with particular properties from the raw signal. Before performing the derivation, the signal may be transformed, e.g., it may be filtered or subsampled.
- *Frequency domain:* Features in the frequency domain are identified by spectral analysis. E.g., the fundamental frequency is extracted from the raw signal by performing a series of spectral analyses.

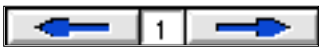
Manual scoring can occur in either domain. Scoring in the temporal domain is explained in this tutorial, and scoring in the frequency domain is explained in the next tutorial.

What You Need to Know Before You Start Scoring

a. Signal Selection

— A signal is "selected" if its 1 - button is black
— Signal selection method 1: Click in the signal
— Signal selection method 2: Click on the 1 - button
— Signal selection method 3: Operate the up and down arrow keys

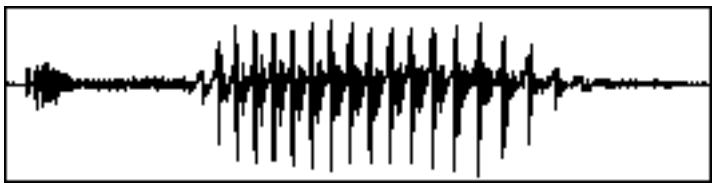
b. Card Selection



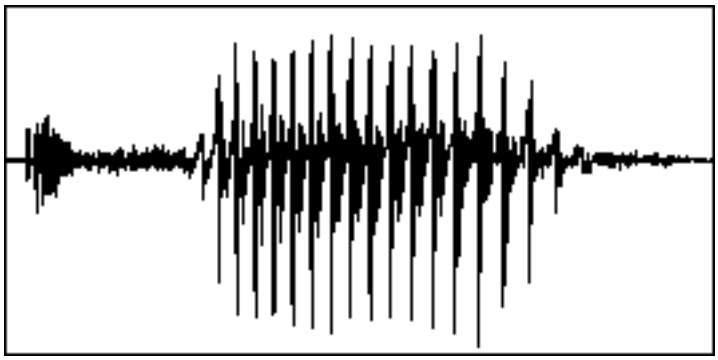
— A signal can be on one of several "cards".
— Card selection method 1: Click on the ◄ or ► button. The number between the two buttons shows which card you're on.
— Card selection method 2: Click on the number between the ◄ and ► buttons. A dialog appears. Type the number of the card you want to jump to. Click "Ok" or press RETURN.

c. Narrow and Wide Signal Displays

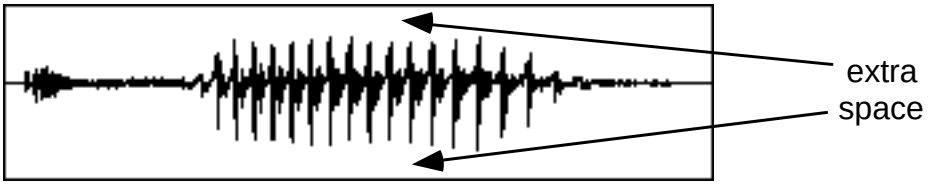
— A signal can be small (narrow), like this:



— Or a signal can be twice as large (tall, wide), like this:



— Finally with both narrow and wide signals, you can leave some extra space above and below the signal:



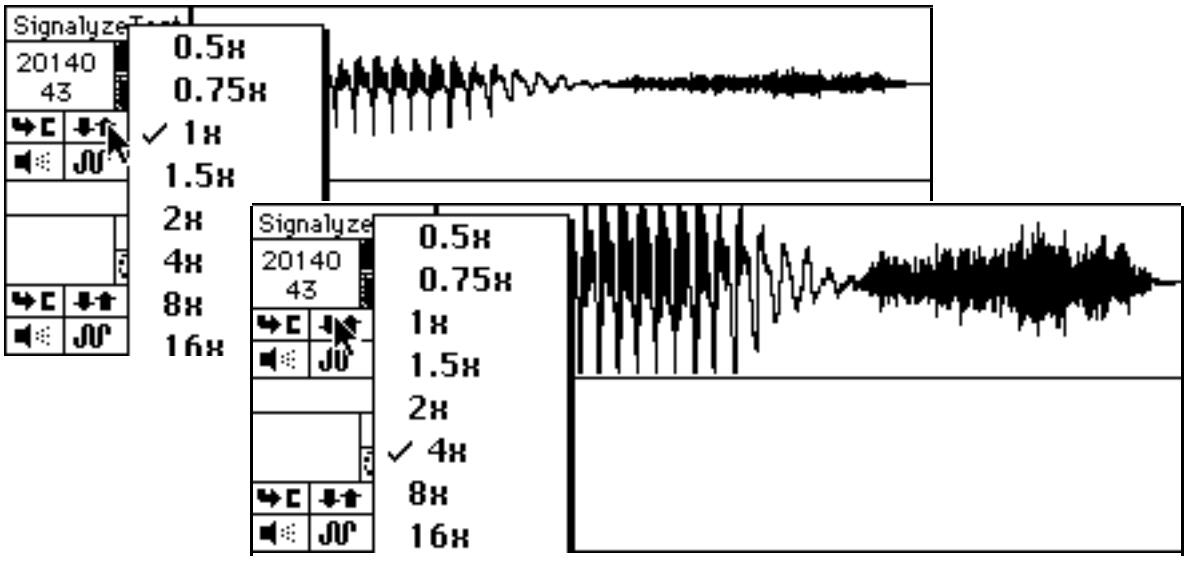
• Use the - button in the Signal Control Panel to toggle between narrow and wide signal displays

— To toggle from narrow to wide, the following tracks must be available:
Top part of display: Uneven-numbered (1, 3, 5, etc.)
Bottom part of display: Even-numbered (2, 4, 6, etc.)

— When toggling from wide to narrow display, the top (uneven-numbered) display will be used.

• To insert an extra space above and below the signal, SHIFT-OPTION click on the replot ( ) button.

d. Vertical Zoom



SHIFT-click on the -button to show popup menu to adjust vertical zoom. This is handy when inspecting the weaker portions of the signal.

e. Moving a Signal to Another Track

— A signal can be in (nearly) any signal track you want

— The restriction refers to wide-display signals: Double-sized signals can only be in an uneven+even-numbered track pair.

— To move a signal to a new track, do the following:

- Click on the - button of a free signal track. A list of current signals appears in the upper left hand corner.
- Double-click on the signal you want to copy. The chosen signal appears in the selected signal track. You now have two *copies* of the same signal.
- Optionally, COMMAND-click on the - button of the original signal. This erases the original copy of the signal, but it does not erase the signal itself.
- You can also move a signal to a different card by this method.

f. Deleting a Signal

Click on the signal's - button. This erases the signal and all of its copies on all cards.

Please remember that *signals* are independent from their *displays*.
You can delete a *display* without deleting its *signal*.
Deleting a *signal* will also delete all its *displays*.

g Compare two Different Zooms

— Zoom the first signal as desired.
— Disable the first signal's - button. It should look like this: .
— If you have labels in your signal, the buttons look like this: and .
— Zoom the second signal as desired.
— The first signal does not move until enabled again by a click on its "enable" ( ) button.

h. Missing Values

"Missing values" are real samples, but they are *non-displayed* samples. When the plotting routine encounters such samples, it ceases to draw the signal line. This makes it possible, for example, to show blank sections for parts of a pitch extraction where no voicing energy was found. "Missing values" are internally represented by the value 32767.

Manual Scoring in the Temporal Domain

a. Taking Duration Measures

- Explicit Method with Clicking

— Zoom the signal as desired.
— If you wish, write a label into the Edit Window, then press TAB.
— Blacken signal segment of interest.
— Click on the duration measure in the General Control Panel (the box labeled "d").
— Press RETURN to add the duration measure to the Text Window.

- Faster Method for Series of Durations

— Zoom the signal as desired.
— If you wish, write a label into the Edit Window, press TAB.
— SHIFT-COMMAND-click at the beginning and at the end of the signal segment of interest.
— The two time indications are shown in Edit Window.
— Press RETURN, or double-click in signal, to add the two time measures to the Text Window.
— Calculate durations with a spreadsheet program by taking differences between the two measurement columns.

- Keyboard Method With "Show Values"

— In Edit>Show Values Options, select only "duration".
— If you wish, write a label into the Edit Window, press TAB.
— Blacken the feature of interest.
— Perform Edit>Show Values, or COMMAND-G.

- Keyboard Method with "Show Statistics"

— If you're collecting statistics other than duration, you may wish to include duration at the same time.
— Select "duration" in Edit>Show Statistics Options.
— If you wish, write a label into the Edit Window, press TAB.
— Perform Edit>Show Statistics, or COMMAND-U.
— This method is slower than the "Show Values" method.

b. Get a Specific Numeric Value

- Locate the Maximum Value in a Signal

— Make sure you've selected a signal (click in it if necessary)
— Place the cursor at the left edge of the desired signal and open Edit>Find in Signal

— The "find" dialog reads like an English sentence. By clicking in the right places, make it read the following:

"In signal <x>, find a value equal to ("=") the maximum value, and that has no specific direction."

— Click on "Find".

— If necessary, zoom the signal segment containing the search value and perform Edit>Find Again.

• Locate a Value Greater than a Given Value

— Suppose you want to find those portions of a signal that exceed the mid-point value of a signal.

— Open Signal>Signal Information and obtain the maximum and minimum values. Calculate the midpoint ((max - min) / 2).

— Place the cursor at the left edge of the desired signal and open Edit>Find in Signal.

— By clicking in the right places, make the find pattern read:

"In signal <x>, find a value greater than (">") the value <midpoint value>, and that has no specific direction."

— Click on Pattern 1. This lets you find the specified pattern with the simple command Edit>Find, or Edit>Find Again, once you have left the dialog

— Click on "Find"

— The first corresponding value is found when exiting. Succeeding values can be found by requesting Edit>Find Again or COMMAND-E.

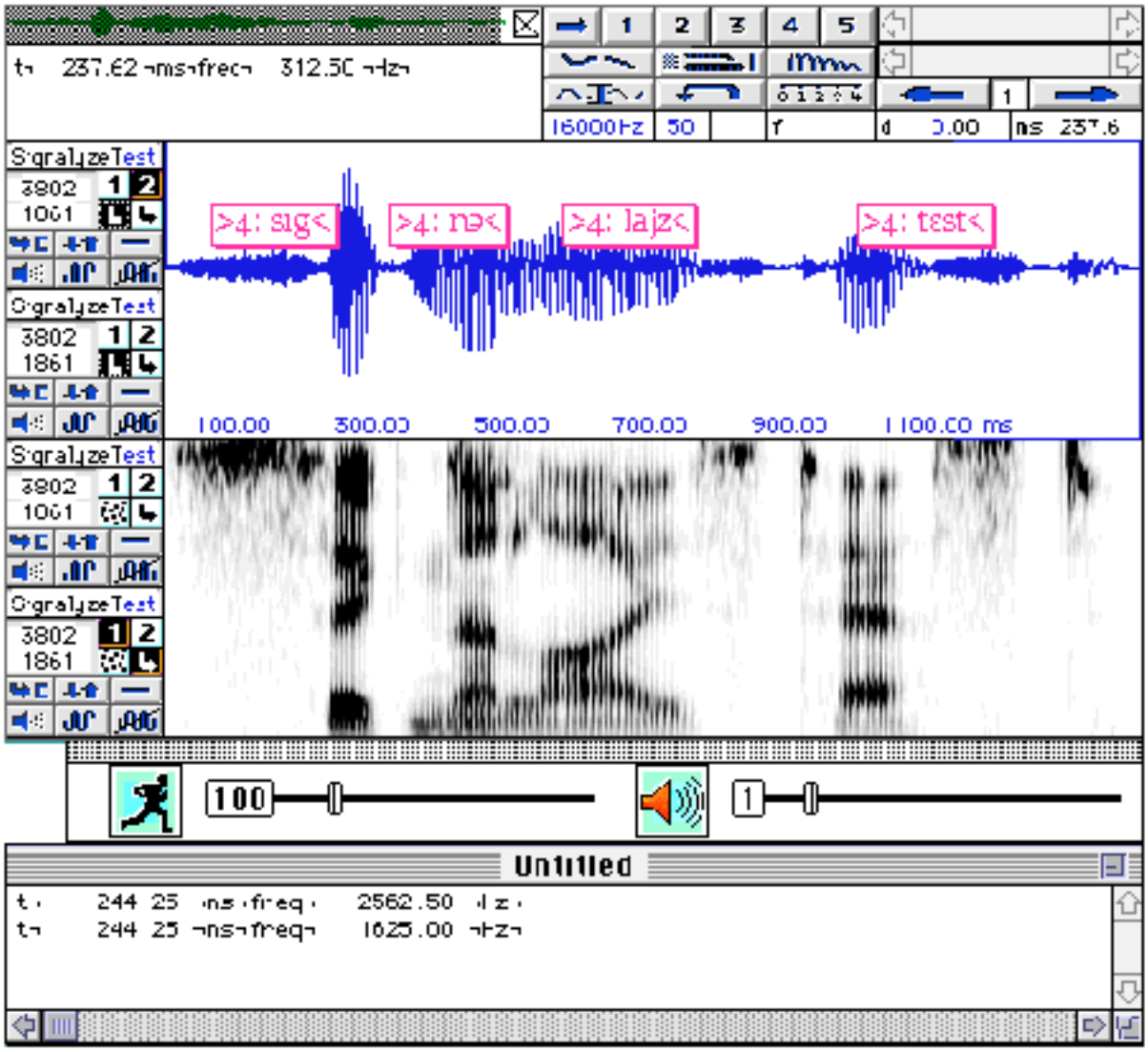
— Two find patterns can be specified. They can be called, respectively, by calling Find/Find Again with

SHIFT LOCK released
SHIFT LOCK KEY engaged

For Macs with Large Screens

Try the following window arrangement for manual scoring:

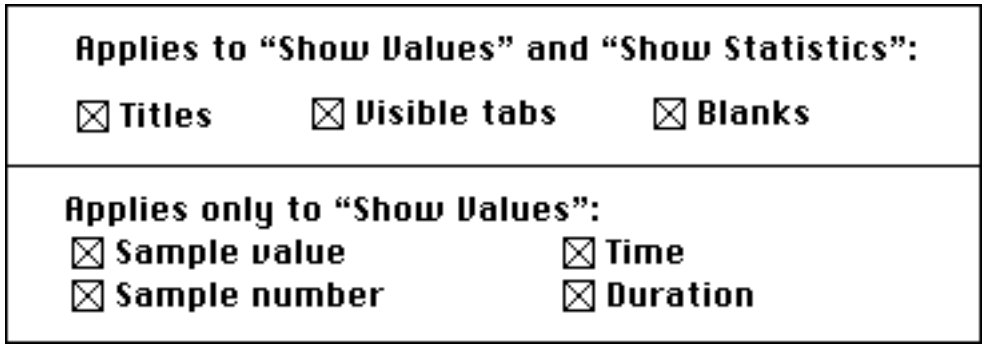
— In Edit>Startup Options, set "Manual Window Sizing".
— Specify 4 displays per card.
— Close dialog and Quit Signalyze.
— Restart the program.
— Resize the Text Window below the Signal Window.



c. Change a Specific Signal Value
— Place the cursor on the desired value
— SHIFT-click on the signal's sample value (the lower of the two numbers in the Signal Control Panel)
— Change the number as desired (limits: -32,766, + 32,766). If you set a value of 32,767, the sample would become a "missing value", i.e., the sample would still be there, but would not be shown in the graph.
- Click "Ok". The signal value is changed.

What Gets Transferred to the Edit Window

By specifying the options in Edit>Show Values Options, the user has full control over what gets transferred to the Edit Window, and from there to the Text Window.



Options in the top half of the dialog specify whether numeric values are surrounded by titles, visible tabs and/or blanks.

Title: An abbreviated title, like "t" for "time indication" or "val" for "sample value". The full list of abbreviations is found in the appendix.

Visible Tabs: The TAB character is visible as a hyphen with a hook (¬) if you've set the "Visible Tabs" option. The TAB looks just like a blank if this option is disabled. The TAB character renders Signalyze text files compatible with statistics and spreadsheet programs. When typing labels, you must add the TAB character yourself. It's not added automatically.

Blanks: This is useful for aligning columns in the Text Window when using an equal-spaced font like Monaco or Courier.

Appearance of numerals: Specified in Signal>Numeric Setup (see explanations below)

The options in the bottom half of the dialog specify which values are transferred to the Edit Window with Edit>Show Values.

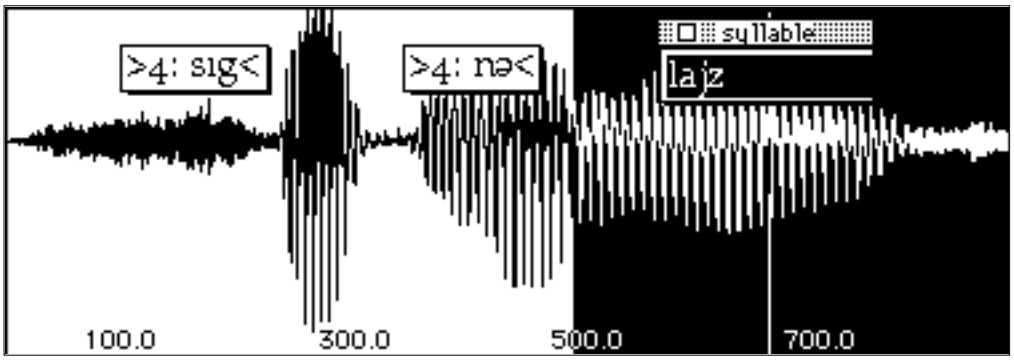
Sample value: The bottom number in the Signal Control Panel of the selected signal

Sample number: The top number in the Signal Control Panel of the selected signal

Time: The time indication from the General Control Panel

Duration: The duration indication from the General Control Panel

Labeling



Labels at level 4 (here, the syllable level). Do CTRL-click to open any of these labels.

An open label, about to mark the blackened signal selection. Use CMD-Y to play the signal selection.

Signalyze makes it easy to place, maintain and search for labels.

Set Label Levels and Create a Label

Be sure you have a signal in a signal track and select the signal. Open Edit>Label Setup.

This dialog lists nine levels, because in Signalyze labels are obligatorily marked by level. That is, for each label, you specify a level, such as "segment", "syllable", "phrase" etc. Each level has its own label color (if you have a color monitor). Labels are marked for their level by a number placed in front of the label name (e.g., "4: sig", means that the label "sig" is marked for the fourth level).

You can also choose a specific font for the labels. If you want to use a phonetic font for labeling, stick to *SigPalFont* (the shareware font supplied with Signalyze). SigPalFont preserves the numbers and angular brackets you need to indicate labeling levels, which is usually not possible with other phonetic fonts.

If you choose the option "Show ms", the exact millisecond position is shown alongside with the label.

You set and edit the *label levels* through Edit>Label Setup. Outside the dialog, you can switch between labels with a popup menu obtained by clicking in the signal with CONTROL-shift.

The *labels themselves* are created by CONTROL-clicking at the desired place in the signal. Once the label is written, close it with RETURN or by clicking on close box. You'll see the label "snap" to the next standard position, and the signal now has an -button instead of the (enable) button. At any time, the label display can be turned on or off by the menu command Edit>Show Labels.

Label Points or Signal Segments

You can label either points in the signal or selected portions of the signal. To label a point, simply click at the desired place before opening the label. To label a segment, blacken the desired segment before opening the label. The labels for selected portions will be placed at the center of the selection and will be marked by angular brackets (">...<").

Each label must have a *different millisecond time-stamp*. If you attempt to place a second label at exactly the same place as another label, you'll get an error message.

While the label is open for editing, you can *play* the signal by doing COMMAND-Y. If you're labeling a selection, and if you've set the audio to play signal *selections* (via Signal>Audio Setup), you'll hear the segment which you're about to label.
position.

Default Label Options

Above, we opened Edit>Label Setup after having clicked in a specific signal. Each signal has its own label setup which is saved in the .lbl file. However, you can also specify a set of default options by simply clicking in an empty signal slot prior to opening the setup dialog. All new signal labels will initially be labeled according to the default setup.

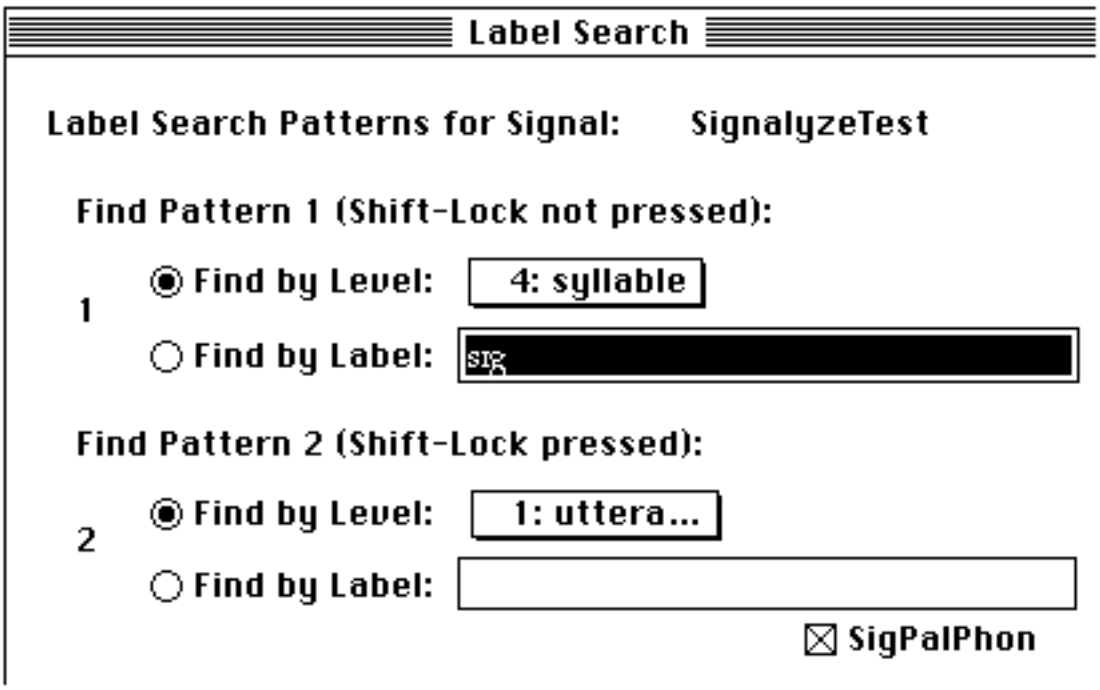
Saving and opening label files

The label information is saved as a TAB-delimited TEXT file with the extension ".lbl". It is stored in the same folder as the signal file. Refer to the Reference portion of this manual (menu Edit>Label Setup) for the detailed structure of label files.

Open File operations on signal files that have accompanying ".lbl" TEXT files cause the label information to be read into Signalyze. *Save File* operations on signals with labeling information automatically saves the labels in the same folder as the signal. Save File operations for signals without labeling information erases whatever label file with the same name that may have existed in the folder. At the present, there is no method for saving labels elsewhere than with the signal.

Searching for labels

You can search for labels using either a given labeling level or the label's name (see Edit>Search by Label). You can specify two separate search patterns. Switch between the two patterns with the SHIFT-LOCK key. If you want to search for a label using the character set of a specific font (e.g., the phonetic font "SigPalPhon") activate the font button at the bottom of the Label Search dialog. The name of the button corresponds to the font you have selected for the signal through Edit>Label Setup.

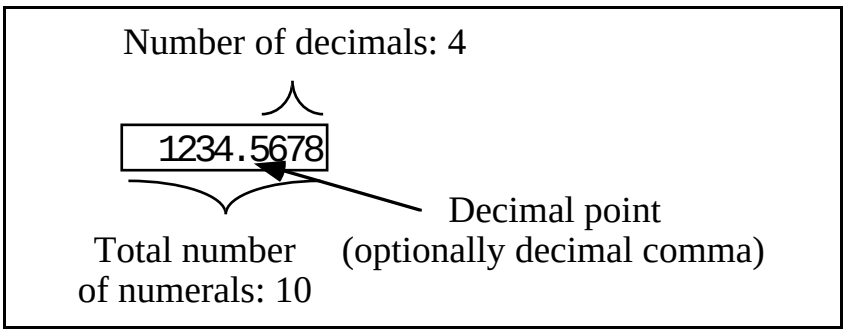


Size Limit of the Text Window

A maximum of 32,000 characters are permitted in the Text Window. While that shouldn't ordinarily crimp anyone's manual scoring style, it can be a problem when trying to read a longer text file generated by another program. The maximum ensures that as much memory as possible is reserved for signals.

Numeric Formats

The Signal>Numeric Setup sets the form in which numbers and signal names are transferred to the Edit and Text Windows.



The Numeric Setup dialog controls a number of aspects of the appearance of numerals:

- **Total number of numerals shown**: If the "blanks" option is selected in Edit>Show Values Options, unoccupied numerals are filled with blanks.
- **Number of decimals**: The numerals to the right of the decimal point or decimal comma.
- **Decimal point/comma**: The symbol used to separate the integer and the decimal portions of the number. People outside English-speaking areas of the world may wish to select the decimal comma option. However, remember that most current spreadsheet and statistics programs do not support the decimal comma standard.

The Numeric Setup dialog also specifies if time indications are in seconds, milliseconds or microseconds.

Signal Statistics

Signalyze can generate a whole series of statistics specified in Edit>Show Statistics. These fall into five main groups:

- Minimum, maximum, delta, ratio, slope.
- Left and right end points, dB and Hz differences.
- Mean, sd, kurtosis, skewness.
- Number of observations.
- Duration.

- ***Minimum, Maximum, Delta, Ratio, Slope***

These operations relate to the minimum and maximum values in a given signal segment. "Missing values" are not counted.

— *Minimum, maximum:* Minimum and maximum values in selected signal segment.

— *Delta:* Arithmetic difference between minimum and maximum. **CAUTION**: This is rarely equivalent to the difference between the two *end point values* of a signal selection.

— *Ratio:* The result of the division of the maximum value by the minimum value.

— *Slope:* Delta/duration, given as: (maximum-minimum)/seconds

- ***Left and Right End Points, dB and Hz Differences***

These operations relate to values found at the two extremes of a signal selection. If one or both of these is a "missing value", the requested operation is not performed.

— *Left and right end point values:* The first values inside the selected (blackened) portion of a signal.

— *dB Difference:* The amplitude difference between the two end point values, expressed as dB. This operation applies only to power and RMS envelopes, and to spectra.
— *Hz Difference:* The difference in Hertz between the two end point values. This operation applies only to spectra.

- ***Mean, sd, Kurtosis, Skewness***

These are standard statistics operations applying to all non-missing values in a selected signal segment.
— *Mean, sd:* Measures of average tendency and degree of variation
— *Kurtosis, skewness:* Measures of deviation from a normal distribution. Kurtosis measures degree of peakedness/flatness and skewness measures deviation from a centered peak.

- ***Number of Observations***

These measures report how many values constitute the selected signal segment.
— N valid observations: Values other than "missing values".
— Total N observations: Total number of observations, including "missing values".
— % valid observations. Percent of total values in selected signal segment that are not "missing values".

- ***Duration***

The duration of the selected signal segment, measured in seconds.

Example: Statistics on a Pitch Extraction

- Load and select the "SignalyzeTest" signal.
- Make sure there are two free display tracks on the current card. If necessary, delete signals with the Delete button ( ).

- Open the Spectral>Pitch Extraction Setup dialog.
- Select the FFT-Comb pitch extraction routine. Set the interval at 5 ms.
- Select "Male Voice" and change the Maximum to 150 Hz.
- Set the Threshold Level to 20%.
- Select Final Output Filter and set it to 8 extractions.
- Select the "dots" and "large track" display options. Deselect "Filter first". Exit with "Do".
- The dialog box closes and a pitch contour is generated.

- Open the dialog box "Signal>Numeric Setup".
- Select 10 numerals and 3 decimals.

- Exit from the dialog box.

- Open the dialog box Edit>Show Values Options.
- Select “Titles” and “Visible Tabs”, deselect “Blanks”.
- Exit from the dialog box.

- Open the dialog box Edit>Show Statistics Options.
- Select “maximum”, “minimum”, “delta”, “ratio max/min”, “slope” and “N valid observations”.
- Exit from the dialog box.

- Blacken (select) the first part of the pitch extraction contour representing “Signalyze”. Listen to the selected part of the signal by COMMAND-clicking on the playback button ( ).
- Perform Edit>Show Statistics or command-U.
- You get a line of information like

```
mx¬135¬mn¬98¬max/min¬     1.378 ¬
dlta¬37¬slope¬   -72.549 ¬valN¬102¬
```

- This means that—
  the maximum value was 135 Hz
  the minimum value was 98 Hz
  the ratio between maximum and minimum was 1.378
  the delta between maximum and minimum was 37 Hz
  the average slope of the descending pitch curve was -72.549 (negative because the curve descends)
  there were 102 valid observations (non-missing values).

These values tend to be useful in evaluating pitch curves numerically.

- Save these values to the Text Window in the usual fashion (e.g., by double-clicking in the signal, or by pressing RETURN).
- If you want to add the values to the current line in the Text Window without a return character at the end, use the ENTER key.

Text Search and Replace

In the Text Window, search and replace operations are possible. They operate in ways familiar from other Macintosh applications:

- Select the Text Window.
- Open “Edit>Find in Text”. Fill in a search string and, optionally, a replacement string.
- Search and replace strings with the appropriate buttons.
- Exit from the search and replace dialog with “Exit”.

Setting the Creator Signature and Saving the Text File

Before you actually save your text file, you may want to set its *creator signature*. This marks the file for its destination program.

For example, if you'd like to analyze your results with Microsoft Excel, you can specify the Excel creator signature (which happens to be "XCEL"). That way you can double-click on the file after leaving Signalyze; the destination program will automatically start up, and the file containing your scoring results will be loaded in. Not all programs maintain this mechanism for text files. If you load Signalyze to work with your text file, specify "BWSW".

- Open the Edit>Preferences dialog box and set the "Creator Signature for Text Files" to the appropriate abbreviation (e.g., "EDIT" for Edit, "MACA" for MacWrite, "XCEL" for Microsoft Excel, "MSWD" for Microsoft Word, "STAT" for StatView, or "BWSW" for Signalyze).
- Save the text File>Save Text or (with the Text Window selected) with File>Save Text As.
- The Text Window can be erased by selecting the Edit>Select All Text command (command-A on the keyboard), followed by the Edit>Clear command (command-B on the keyboard), or by choosing the File>New command.
- The Text Window can also be saved with the Signal Window in front (COMMAND-T). This lets you continually save your results while scoring in the Signal Window.

Adjusting the Text Window Width

In spreadsheet fashion, lines in the Text Window can be longer (but never shorter) than screen window width. Set the wrap-around limit of the Text Window in Edit>Preferences ("Text Window Width").

Fonts and Font Sizes

At program startup, the Text Window font is set to Monaco 9, and the Signal Window is set to Geneva 9. To change fonts:

- Open the font and size menus through Window>Optional Menus

- Change font and size as desired. Your choice of font and size apply to the *entire* contents of the Text Window.
- Font and size selections can also apply to the Edit Window in the upper left corner of the Signal Window.

Compatibility with Spreadsheet Programs, Statistics Programs, Word Processing Programs, Other Computers

Results files created in the Text Window are directly compatible with spreadsheet programs like Excel, or with statistics packages like StatView or SYSTAT, since it is a standard TEXT file containing only ASCII character-numerals and Tabs, and no formatting information.

Even if the text file was not saved with the creator signature of the destination program, you may still be able to read it. Start the program in question, and try using the program's *Open* or *Import* commands in the File menu. Some programs let you choose an "ASCII" "Text" or "Numeric" option in their Open File dialog box. Select the file containing your scoring results and load it into the program. Other programs read text files with the command "Import Text".

Your file can be edited further with standard word processors (MacWrite, Microsoft Word, etc.). Items are separated by Tabs, which causes them to be correctly inserted into columns. The word processor's TAB settings may have to be adjusted to line up the numbers in the desired fashion.

Files can also be transferred to other computers. **CAUTION**: international characters or special symbols tend to differ from one type of computer to another. Some destination computers only read the characters of the lower ASCII set (ASCII 32-126). In those cases, it may be wise to stick to a reduced ASCII set.

Tutorial 5: Spectral Analysis

This chapter is concerned with the tools for performing signal analysis in the frequency domain, i.e., *spectral analysis.* Some basic notions are introduced, and the user is guided through a number of representative examples of how spectral analysis is applied to speech signals.

General Information

Signalyze provides a range of spectral analysis tools: FFT ("Fast Fourier Transform"), cepstrum, LPC, autocorrelation and temporal structure analysis .

The FFT

Currently the most common spectral analysis tool is the FFT. In this approach, the curves in a "window" of signal values are identified and their sinusoidal components are distinguished. Based on the strength of each sinusoidal component, a frequency spectrum is built up. For some straightforward mathematical reasons (basically, that you need at least two points to describe a curve segment), an FFT routine can only reliably extract frequency information from a signal for the *first half of the recorded frequency domain*. This half-frequency is known as the *Nyquist Frequency*. If you recorded a signal with a *sampling frequency* 20 kHz, the FFT can only provide frequency information for the first 10 kHz. 10 kHz is thus the Nyquist frequency of a signal sampled at 20 kHz. If you recorded at 10 kHz, you can get information for the first 5 kHz.

In performing the spectral analysis, an FFT can be attuned more strongly to *wide-band* or more strongly to *narrow-band* events. With wide-band analysis of an acoustic speech signal, the vocal tract resonances (or "formants") are emphasized. With narrow-band analysis, the signal's glottal pulses (or "fundamental frequency") and their harmonics are brought to the fore. The terms "wide band" and "narrow band" have come down to us from the time of analog filters which selected either a wide (about 150 Hz) or a narrow (about 50 Hz) band of frequencies for analysis. In an FFT analysis, wide and narrow band filters are simulated by taking smaller or larger proportions of an analysis window into the FFT routine. In Signalyze, the wide band analysis uses about 8 ms out of each 25-ms window, while the narrow band analysis uses the full 25-ms

analysis window. (The extra-wide band analysis uses 5 ms). The remainder of the 25-ms window is set to zero.

To show the formants of women and children, Signalyze has an *extra-wide band* selection. Because their voices are higher, frequencies that contribute to formant resonances lie further apart, and a wider filter is required to make them maximally visible.

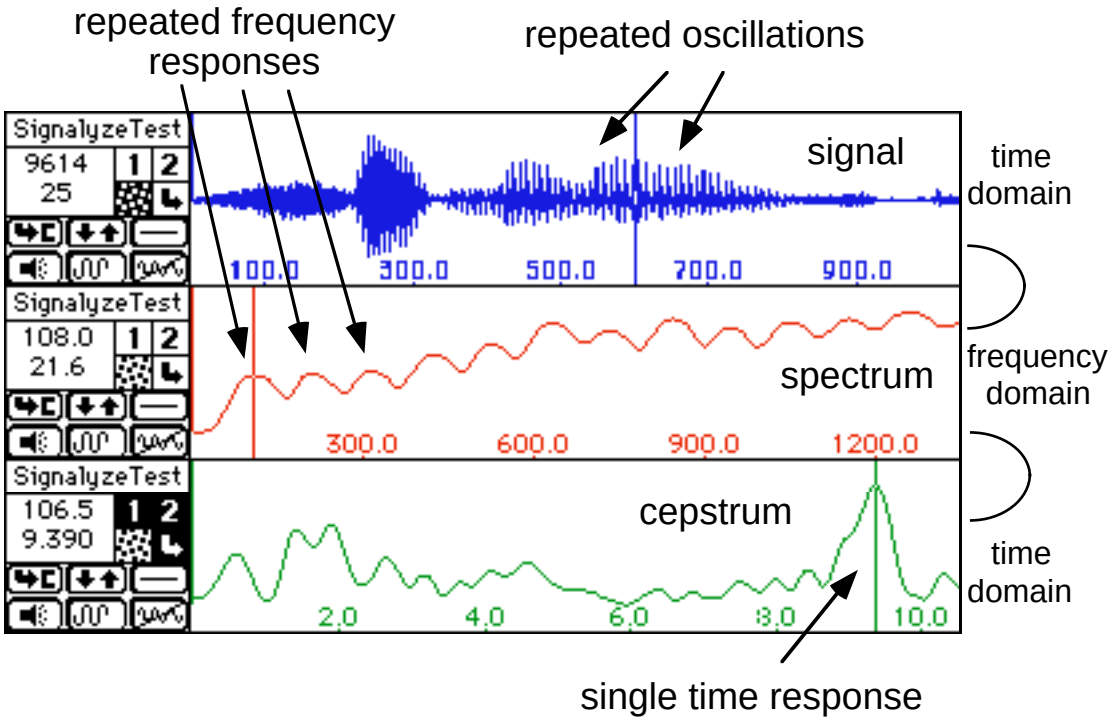
The FFT is used for spectra and cepstra, spectrograms and cepstrograms, in LPC, and in some pitch extractions.

Advantages of FFT in pitch analysis: No need to circumscribe the frequency range very closely. Rather reliable in normal speech. Good distinction of voiced and unvoiced periods.

Disadvantages of FFT in pitch analysis: Relatively slow. Also, the FFT window imposes a filter that smoothes out rapid voice modulations.

Spectrum vs. Cepstrum

The term "**ceps**trum" was created by reversing the first four letters in "**spec**trum", to indicate that this type of analysis is the inverse of spectral analysis. Here's a short explanation of how a cepstrum works:



One way of looking at spectral analysis is to say that it captures *repetitive* elements in an analysis window. A basic FFT, for example, detects the large, *repeated signal oscillation* characteristic of the glottal pulse in the raw signal (top panel) and interprets this oscillation as a *single frequency* (the fundamental frequency, middle panel). In the same way, smaller,

repeated oscillations of given cycle lengths are each summarized as a harmonic (also seen in middle panel).

Since the fundamental and its harmonic frequencies tend to be equally-spaced, the FFT algorithm can then be applied a second time to capture these repetitive frequency elements (in the middle panel) and display the result as a *single response* in a *new time domain* (bottom panel). The new time domain is generally measured in ms, but it is also sometimes called the "Quefrency", to indicate that it is the inverse of "frequency". As can be seen, this approach incorporates both the fundamental frequency and its harmonics into a single number, the cepstral response. Cepstral analysis is thus a fairly reliable way of obtaining a single fundamental frequency measure from a signal incorporating both fundamental frequency and its harmonics. Since in speech, the fundamental frequency is often imprecise and difficult to locate with a simple FFT, cepstral analysis is often employed as a more reliable means of identifying the fundamental frequency.

Autocorrelation

An autocorrelation analysis extracts frequencies by determining the degree of similarity between two curves. Since two successive glottal pulses in a speech signal look very similar (and thus have similar numeric values), the correlation between them is very high. The fundamental frequency (or "Fo") can thus be determined from the strongest correlation peak found in a series of comparisons of successive signal segments.

In Signalyze, autocorrelation is used for one of the three pitch extraction algorithms, as well as in LPC analysis.

Advantages of autocorrelation in pitch extraction: Rather reliable over a wide range of conditions, including noisy background. Responds only to fundamental frequency and ignores harmonic frequencies. Good distinction of voiced and unvoiced periods.

Disadvantages of autocorrelation in pitch extraction: Rather slow. Also, the autocorrelation window imposes a filter that smoothes out rapid voice modulations. May be unsuitable for analysis of voice disorders.

Linear Predictive Coding (LPC)

LPC is another means of extracting frequency information from a signal. It combines both autocorrelation and FFT approaches to abstract a somewhat idealized frequency spectrum from the signal.

Spectral analysis by LPC first performs an autocorrelation analysis, then abstracts the autocorrelation information into a small number of points, and finally performs an FFT on these points to obtain the spectral information. The "small number" is that given by the LPC's "order" setting. If the order number is well fitted to the signal, a surprisingly

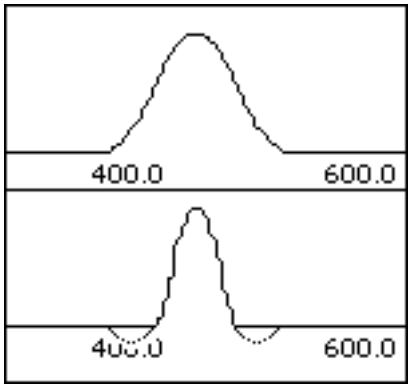
clean image of the vocal tract resonances (formants) can be generated by this method.

CAUTION: USE LPC WITH CARE. If the wrong order number is selected, LPC formant measures often diverge substantially from narrow-band or wide-band formant measures. It is thus important to "play around" with the order number before doing a series of LPC measures. Despite these disadvantages, LPCs can be useful to distinguish two closely situated, adjoining formants (as F1 and F2 in the vowel /a/), especially in cases where wide-band analysis cannot show the distinction. Here are the standard recommendations for LPC analysis:

- Respect *anti-aliasing constraints* when sampling the signal. (Nearly all current sound digitizers apply an anti-aliasing filter to the signal prior to sampling. As a result, the user doesn't generally need to concern himself with this issue.)
- Use *sampling rates* as low as possible, i.e., a bit more than twice the highest expected formant frequency.
- *Order (coefficient) number:* Take 2 orders per expected formant, plus 2 or 3. The 2 or 3 extra orders model the laryngeal pulse. First approximations generally work out as follows: for males, take 2 orders, for females 1 order per kHz below the Nyquist frequency, plus 2 or 3.
  Example: Suppose you want to see four formants in a male voice. The highest expected formant frequency is 5 kHz. Sample at 13 kHz, with an anti-alias setting of 5.2 kHz (40% of 13 kHz). There are 6 kHz below the Nyquist frequency of 6.5 kHz, so calculate 2*6=12, add 2 or 3 to get 14 or 15.
  If you use too many coefficients: You get spurious peaks around harmonic frequencies.
  If you use too few coefficients: Several peaks coalesce into broad "mountain ranges".
- Since the number of coefficients (orders) is so small, the *chances of error* in the model are relatively high. For this reason, LPC extractions are quite often "strange" (no peaks, weird-looking peaks, or peaks in patently wrong places). Nudge the cursor over just a bit and try again. Chances are you'll do much better.

Cone Kernel Techniques

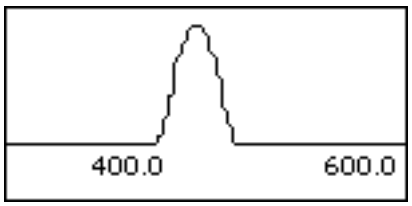
There exist a number of new spectral enhancement techniques. Of these, cone kernel techniques can often improve formant visibility considerably.



Frequency response to a sine tone. Above: Narrow band FFT. Below: Narrow band cone kernel, basic cone. The dotted-point negative values produced by the cone kernel routine are set to zero for the actual display (see figure below).

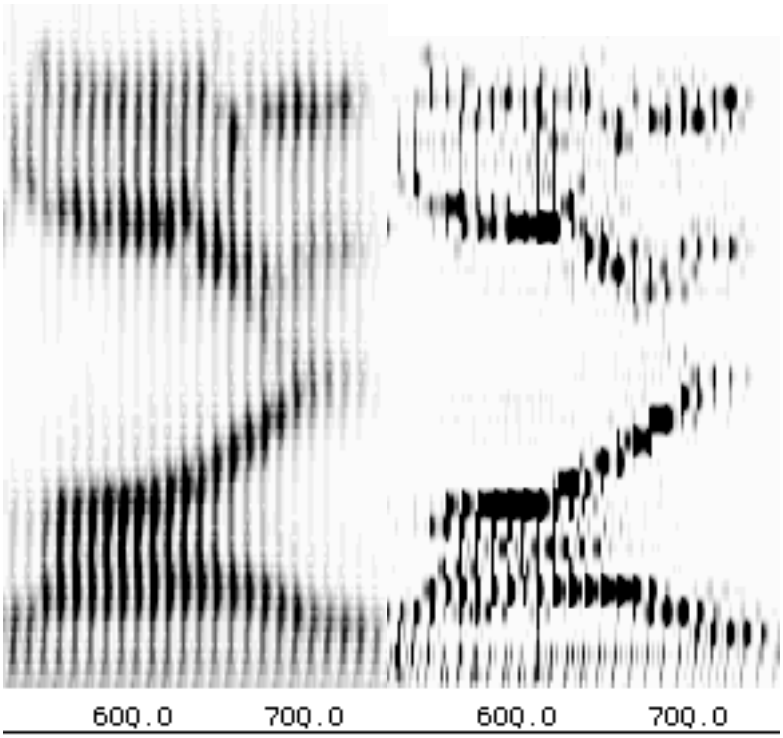
Cone kernels work by redistributing the signal information by means of a special kind of autocorrelation which is applied prior to the FFT. This has the net effect of imposing a sort of "Mexican hat" on the time-frequency space. By contrast, a spectrum coming out of the FFT looks something like the upper graph in the figure above. The spectrum is taken from a narrow-band FFT performed without pre-emphasis on a sine tone signal.

In Signalyze's real-world implementation, the cone kernel is shown minus the dotted negative excursions to either side of the cone. It is customary to suppress them, since they do not contribute to useful information in speech analysis. So what the user sees is the figure below.



The narrow-band cone kernel spectrum as it appears to the user.

It can be seen that the cone kernel provides a much narrower spectrum than does the simple FFT. In addition, the cone kernel techniques provides much improved precision in the frequency ***and*** temporal domains. This is why the technique goes by the name of "cone kernel": Viewed in the three-dimensional time-frequency-amplitude space, the routine acts as if it placed a *steep cone* upon the usual, broad-shouldered FFT frequency response.



Spectrograms of the diphthong [aj] from the signal "Signalize Test", at left with a normal wide band setting and at right with the wide band cone kernel option.

The effect is quite striking. In the figure above, broad-band details from the diphthong [aj] found in the signal "Signalyze Test" are compared over FFT and cone kernel techniques. It can be seen that the cone kernel spectrograms are much better defined than the usual FFT spectrograms. There is, of course, a price to pay: Not only are cone kernels slower to calculate, but *the dB information obtained from cone kernel spectra is not reliable.* The signal has been so thoroughly transformed that the spectrum's different amplitude levels no longer show reliable relationships.

There are a few more caveats. Articles on the cone kernels usually show beautifully smooth cones. However, real-world sampling constraints usually give a much cruder approximation to the cone. This is particularly so when one chooses the double-diamond routines that work on very short (128-point or 256-point) input data (double diamonds also show considerable ripple in the surrounding frequency domain—use them with care). Wide-band spectrograms (256-point input with the basic cone) give cruder cones than narrow-band spectrograms (512-point input with the basic cone), and as can be expected, whole-number FFTs give less precise results than floating-point FFTs. In fact, the whole-number FFT gives such unreliable frequency results (divergences of as much as 80 Hz!), that *users should always use the floating-point FFT with cone kernels,* except when doing spectrograms on Macintoshes where a floating-point FFT is just about an impossible proposition (i.e., SEs, MacPluses).

So to summarize: (1) If ever possible, use the floating-point FFTs with cone kernels, (2) use pre-emphasis and smoothing as usual, (3) generally use the basic cone routine, (4) do not measure dB values, and (5) switch between narrow- and wide-band routines as you would in normal, FFT spectral work.

Temporal Structure Analysis ("TSA")

The last major spectral technique considered here concerns the extraction of the fundamental frequency. TSA attempts to identify strong falls and rises characterizing the onset and termination of large oscillations found in voiced sections of acoustic speech signals. On the basis of the time delay between successive strong fall-rise structures, the frequency of the glottal pulse (Fo) is determined. In Signalyze, TSA is used in one of the three pitch extraction routines.

Advantages of TSA: Very fast. It follows voice modulations quite closely. There is an extraction every voice period. It is insensitive to divergences between fundamental frequency and harmonic frequencies, since it disregards the harmonic information.

Disadvantages of TSA: Unless the frequency range is tightly specified, obtained values are likely erroneous. Also, harsh or irregular voices play havoc with the identification of fall-rise patterns.

Some Typical Spectral Analyses

Spectrum Button

Spectrogram Button

Pitch Extraction Button

- ***A Narrow-band Spectrum of a Vowel***

— Make sure you have at least four empty signal tracks on the current card.
— Load a signal with Open Signal.
— Select default scales by clicking on the scales button ( ).
— Place the cursor in the middle of a vowel.
— Do Spectral>Spectral Analysis Setup.
— Select:
Full range
Narrow band (25 ms window)
Pre-emphasis
Smoothing
Large track

— Click on Spectrum/Cepstrum.
— A spectrum is drawn.
— Click in spectrum. The corresponding frequency is shown as the top number at left. The relative amplitude (dB) is given as the bottom number at left.
— Click on the frequency and amplitude readouts to transfer them to the Edit Window.
— Erase the spectrum with the Delete button ( )
— Experiment with Half and Quarter range. At smaller ranges, you should be able to see the fundamental frequency and its harmonics.
— Experiment with selecting the log scale. The log scale lets you see the fundamental frequency more easily.
— You can get additional spectra by placing the cursor in different places in the signal and by clicking on the Spectrum button.

• ***A Narrow-band Cepstrum of a Vowel***
— Same preparation as for the spectrum
— In Spectral>Spectral Analysis Setup, select:
Full range
Narrow band (25 ms window)
Pre-emphasis
Cepstrum and Low Pass
Smoothing
Large track
— Click on Spectrum/Cepstrum.
— A cepstrum is drawn. You should see a large peak somewhere between 2.5 and 10 ms.
— Click in the peak. The corresponding frequency is shown as the top number at left. The Fo cycle duration is given as the bottom number at left.
— Erase the cepstrum with the Delete button ( ).

• ***A Wide-band Spectrum of a Vowel***
— Same preparation as for narrow-band spectrum.
— In Spectral>Spectral Analysis Setup, select:
Full range
For men's voices: Wide band (8 ms window)
For female or children's voices: Very wide band (5 ms)
Pre-emphasis
Smoothing
Large track
— Click on Spectrum/Cepstrum.

— A spectrum is drawn. You should see a series of peaks corresponding to formants. If not, move the cursor a bit to the right or the left and start over.
— Obtain frequencies and amplitudes as in narrow-band spectrum.
— Experiment with the inverse filter. It quite often improves the visibility of formants.

- ***Save Decibel Values from a Spectrum into a File***

— Draw a spectrum as desired.
— Select the spectrum
— In Signal>Numeric Setup, specify the number of decimal values for the desired output
— In "Save Signal As", select ASCII (TEXT) format.
— A dialog appears, giving the option to save in decibel form.
— If the decibel option is chosen, the spectrum is saved according to the numeric format specified before.

- ***Narrow-band Spectrogram on Signal Window***

- Make sure you have two signal display tracks available in the Signal Window.
- Set scales and/or x-y gridlines with Signal>Scale Setup.
- Select the desired signal segment. If no signal segment is blackened, the entire signal width is selected
- In Spectral>Spectral Analysis Setup, select:
  Narrow band (25 ms window)
  Quarter range
  Pre-emphasis
  Smoothing
  Colors: as desired and as possible on your monitor
  Repeat every other line
  Separate window off
- Click on Spectrogram/Cepstrogram
- A narrow-band spectrogram is drawn in the next available two-track space below the selected signal segment.
- Adjust the darkness with SHIFT-click in the numbered rectangle below the Abort ( ) button.
- SHIFT-COMMAND-click to get the indications of time and frequency at clicked position.

CAUTION: The frequency indications obtained in this manner are rather imprecise. They simply give a general idea. Use spectra to get more precise frequency measures.
- You can get additional spectrograms by clicking on the Spectrogram button ( ).

• ***Wide-band Spectrogram on Separate Window***

- In "About Signalyze", verify that you have enough calculation memory to proceed. To start off, you need about 400 k to run the routines (more on the larger screens). On top of that, you need the following extra amount for each spectrogram window:
  -> With 256 colors/grays and a 21-inch screen, 200 k.
  -> With 256 colors/grays and a 13-inch, 100 k.
  -> With 16 colors/grays and a 13-inch, 60 k.
  -> With 2 colors, you needn't worry.
- If you have a color/grayscale Mac, choose the monitor setting in Monitors Control Panel accessory.
- If necessary, increase calculation memory with Edit>Startup Options.
- Set scales and/or x-y gridlines with Signal>Scale Setup.
- In Spectral>Spectral Analysis Setup, select:
  Wide or Very Wide-band (8/5 ms window)
  Full range
  Pre-emphasis
  Smoothing
  Color or grayscale with a color/grayscale Mac
  (the program automatically selects B/W with black/white Macs)
  Every line
  Print title
- Click on Spectrogram/Cepstrogram

• ***Get Formant Spectrum via LPC***

- If the signal is not sampled at 10 kHz, select it and resample it with the following settings in Transformation>Complex Transformation Setup:
  Output sampling frequency: 10000 Hz
  Output bit width: as in input
  Output bias: as in input
  Output numeric type: as in input
- Click in vocalic portion of 10 kHz signal
- Do Spectral>Spectral Analysis Setup
- Set range to "Full range"
- Set the "Band" to LPC
- Specify "Order 14"
- Specify "Large Track"
- Click on Spectrum/Cepstrum
- If the output contains ill-defined peaks, try moving the cursor somewhat to the left or to the right
- Perform a wide or very wide band analysis (as appropriate), and compare outputs. Peaks should correspond quite closely.

- ***Get Formant Spectrogram with LPC***

- If you have a fast Mac (or lots of time), try running a spectrogram on a separate window with the same LPC settings.
- Especially on color- and grayscale monitors, a rather striking LPC-spectrogram is drawn.
- You can turn the title off with the "Print title" option.
- **CAUTION:** On slow Macs, stopping the LPC may take a while, since the routine checks for aborts only every 10 pixel lines.

- ***Default Pitch Routine: FFT-Comb***

- A signal without excessive background noise and good frequency response below 800 Hz is required for this analysis (perform a spectrum first if necessary).
- Make sure there are at least two free signal tracks below signal (uneven and even-numbered, respectively)
- Select signal and open Spectral>Pitch Extraction Setup
- Set the following parameters
  - FFT-Comb
  - A new spectrum every 5 ms
  - Expected frequency range for male, female or child's voice
  - Initial threshold of 20%
  - Set output filter to 8 extractions
  - Dots and Large Track
- Click on "Do".
- A pitch extraction is calculated
- Frequency values are shown as the lower number in the Signal Control Panel.
- Lower the threshold percentage if the routine suppresses many vocalic portions of the signal. Raise it if it shows too much response in non-vocalic portions of the signal.
- To increase speed, experiment with setting the spectrum interval to 10 ms. With the interval at 10 ms, the output filter should be set to 4 extractions in order to preserve temporal precision.

- ***A Fast Pitch Routine: Temporal Structure Analysis***

- A signal without excessive background noise and a rather "even" voice is required for this analysis (small intonation range)
- In Spectral>Pitch Extraction Setup, set the following parameters:
  - Temporal Structure Analysis
  - Appropriate frequency range. This range should not generally exceed one octave, i.e., the maximum frequency must not be more than double the minimum frequency
  - Initial threshold: 15%
  - Initial output filter: 16 extractions

Dots and Large Track
- A fast pitch analysis is performed
- Play around with the threshold and the output filter to see if you can improve your results.

• ***A Pitch Routine for Noisy Signals: Autocorrelation Analysis***
- A voice signal of nearly any definition can be used.
- In Spectral>Pitch Extraction Setup, specify the parameters as in FFT-Comb, except for the following: "Autocorrelation".
- After a while, a pitch extraction is drawn.
- Play around with the threshold and the output filter to see if you can improve your results.

• ***Calculating the Long-Term Averaged Spectrum***
— The long-term averaged spectrum represents an average of the same FFT-values that are used to draw the spectrogram. Averaged spectra can be drawn up indifferently from spectrograms, cepstrograms or LPC-grams.
— Blacken the segment of interest in the signal. Typically, only similar-looking (e.g., only vocalic, or only fricative, etc.) material is averaged.
— In Spectral>Spectral Analysis Setup, make your selections of band width, routine, range, etc. Select "Averaged spectrum".
— After the spectrogram is drawn, the corresponding averaged spectrum is calculated and displayed in the next available signal track.
— The averaged spectrum can be saved like a normal signal. With the ASCII format, an averaged spectrum (but not an averaged cepstrum or averaged LPC) can optionally be saved to a file in decibel values.

Notes on Spectrograms

Resolution, Speed and Memory Tradeoffs
High Resolution Spectrograms (whether in color or not) quite naturally gobble up lots of memory, but you can minimize memory use by a judicious selection of colors or grays.
• 256-color/gray spectrograms are the prettiest and give the best laser reproduction. But they take the longest to draw and they consume the greatest amount of memory (e.g., about 200 k of calculation memory on a 21-inch screen). Use this standard for publication-quality spectrograms.
• 16-color/gray spectrograms are nearly as good as 256-color/gray spectrograms, but they are quite a bit faster and use only a quarter as much memory (e.g., about 70 k on a 13-inch screen). This is generally a satisfactory standard for manual scoring.

- Black-and-white spectrograms use standard Macintosh dithering to achieve up to five levels of gray. They are the fastest, and they use hardly any memory (e.g., about 20 k on a 13-inch screen). Admittedly, the coarseness of their features may pose problems for manual scoring.
- The greatest increase in speed comes through an increase in processor power and speed. A M86030 processor in a IIcx crunches through the spectrogram much more slowly than a M68040 in a Quadra, for example. Some acceleration cards give spectacular results (e.g., 9 seconds for a 256-color and 500-pixel spectrogram obtained with a super-accelerated Power Macintosh).

Spectrogram Colors

If you don't like the spectrogram colors, you can set them yourself via *Edit>Spectrogram Palette*. This menu command only works if you have a high resolution monitor.

Spectrogram Darkness

The general darkness level influences all types of spectrograms: those in black/white, in grays and in colors. There are three ways to adjust this level:

- SHIFT-click on the darkness indicator in the Information Bar (the number below the Abort -button).
- Adjust the number in the Spectral>Spectral Analysis Setup dialog.
- While drawing a spectrogram on a separate window, a special darkness control appears. The darkness can be changed in mid-course. After modification, the spectrogram is redrawn entirely with the new darkness level, wrapping around the right margin in the process.

Key-click Combinations for Read-outs

This works only with spectrograms in the Signal Window:

- Spectrogram time and frequency: SHIFT-COMMAND-click in the spectrogram.
- Erase Edit Window: SHIFT-OPTION-click in spectrogram.

Speed Demons. High spectrogram display speed is a key element in fast signal analysis. Here are a few ways of getting faster spectrograms:
(1) As much as possible, do without the **log setting**. Log should only be used in narrow band spectrograms and primarily in normal definition spectrograms, if used at all.
(2) **Wide band** spectrograms are substantially faster than narrow band spectrograms.
(3) **Black-and-white** spectrograms are faster than high resolution spectrograms, especially when drawn directly into the signal window.
(4) In high resolution spectrograms, **16-grays/colors** are faster than 256 grays/colors.
(5) Spectrograms with many white areas are quite a bit faster than darker spectrograms, so keep the **darkness level** as low as reasonable.
(6) **Inverse color** is a bit slower than regular color.
(7) **Some accelerator boards** can bring a substantial speed improvement over unaccelerated Macintoshes.
(8) More **power** equals generally more speed.

Recommendations for Spectrogram Settings

	Narrow Band	Wide Band	Very Wide Band
Black/White			
Shading	4 tone	5 tone	5 tone
Plotting	Repeat	Every or Repeat	Every or Repeat
Range	Quarter	Full or Half	Full or Half
Pre-emphasis	On	On	On
Log*	Off	Off	Off
Color/Grayscale			
Plotting**	Repeat	Every or Repeat	Every or Repeat
Range	Quarter	Full or Half	Full or Half
Pre-emphasis	On	On	On
Log*	Off	Off	Off
Spectra			
Pre-emphasis	Off	On	On
Log*	On	Off	Off
Cepstra			
Range	Full or Half		
Low pass	On		
Pre-emphasis	Off		
Log	Off		

* Rarely needed. If used at all, use primarily in narrow band analysis.
** Plot Every Line with full range of higher sampling frequencies (>40 kHz).

Tutorial 6: Multiple Signals and Signal Editing

This chapter explains how to manipulate several signals, how to change signal appearance, how to align signals, and how to change their title. Also, there's a section on editing signals with "copy", "cut" and "paste" commands.

More Operations on Multiple Signals

Time Locking and Synchronous Cursors

Zooming and signal shift operations operate generally in synchronous manner for all signals in the program.

- Disable synchronicity for a given signal via the "enable" button ( ). When the button appears like this: , the signal stays stationary. This lets you compare two different time segments in two signals.
- To toggle between individual and synchronous cursor operation, activate the menu command Signal>Synchronous Cursors.

What All The Signal Buttons Are For

First and Second Signal Buttons

Destination Track Button

- *First Signal Button:* Shows which signal is selected. Click on it to select a signal.
- *Second Signal Button:* Indicates which signal serves in an operation involving two signals (for some examples, check out the Transform>Simple Transformations Setup).
- *Destination Track Button:* Lets you select where the next signal is to be loaded. Click in any empty track to turn on its Destination Track Button, or click on the desired button.

The Logic Behind Signal Placement

Here is the full logic used by Signalyze to assign a new display track:

if (the track indicated by the Destination Button is empty)
take the track;
else if (the signal is generated from another signal [e.g., transformation or pitch extraction] or the "Next Available Display" option is selected in Signal>Plot Setup)
find the next empty track;
take the track;
else if (the signal is loaded in from disk and the "New Card" option is selected in Signal>Plot Setup)
find the first card with an empty first track;
take the track;

In essence, this means that
1. The Destination Track Button has highest priority.
2. Signals generated from existing signals are usually shown on the same card.
3. Newly loaded signals are shown on a new card if the user wants that.

Vertical Zoom and MinMax Setting

The user can adjust a signal's vertical zoom, and — independently — its MinMax setting. Both have an influence on how the signal is shown on the screen.

MinMax: When a signal is first loaded in, its maximum and minimum values are determined to scale the signal to the display track. These values determine the darkness of the spectrograms and many other signal calculations. The most remarkable aspect of MinMax scaling is that weak signals appear "blown up" and suddenly fill the entire signal track. A more subtle aspect is that this scaling is based on a calculation of the signal's true bit width, and that a weak signal recorded at 16 bits may in fact have only a bit width of 15 or 14 bits.

Vertical zoom: At any time, the user can "zoom into" the signal to inspect weaker segments or "zoom out of" the signal to display it without conflict with the time scales (SHIFT-click on the -button). The vertical zoom operation has no effect on any signal operation.

- ***MinMax by Clicking***

- Turn off scales.
- Start the process with Signal>Set MinMax or with COMMAND-M.
- Follow the instructions. After each message appears, wait till the message disappears and then do as instructed. The duration of the message is fixed as "Pause Length" in Edit>Preferences.
- Click at the bottom and at the top of signal feature of interest, as instructed.
- The signal is shown with the specified vertical zoom.
- To return to normal min/max setting, activate the menu command Signal>Original MinMax.

- ***MinMax via the Information Dialog***

- Perform Signal>Signal Information.
- Set the signal minimum and maximum to desired numeric values.
- Exit with "Save".
- To return to normal min/max setting, activate the menu command Signal>Original MinMax.

- ***Disable the MinMax operation***

So see the signal just as it was recorded at 16 bits by another program, you can disable the MinMax operation usually performed when opening the signal by activating the "Force 16 bits" option in File>File Setup. Signals will then have maxima and minima according to the 16-bit standard (±32 766).

The Signal Information Page

Here's some more information on the Signal Information dialog.

At the left, the dialog shows the signal's most recently specified *card* and *track location*. Usually, that's the location where the signal was first shown, but if you've established a second display track for the signal, it shows the location for the more recent display.

Reg indicates if the signals is a "regular" (raw or transformed) signal, or if it is a derived signal. Derived signals have special display properties that are permanently lost when they are converted to regular signals by clicking in the "Reg" checkbox. For example, spectra lose the dB information and instead show the underlying FFT output after conversion to a "regular" signal.

The *signal length* is given in bytes.

Signal names and sampling frequencies are given in edit boxes which you can change as you wish. If you change the sampling frequency, sound signals will sound differently in sound reproduction.

The *Minimum and Maximum values* can be changed to resize the display height. This is particularly handy if your signal contains just a few abnormal values.

8-bit signals range generally from 0 to 256 and have a mean of about 128. Their mean value tends to be around 128.

12-bit signals generally have a minimum value of 0 and a maximum value of 4096. Their mean value tends to be around 2048.

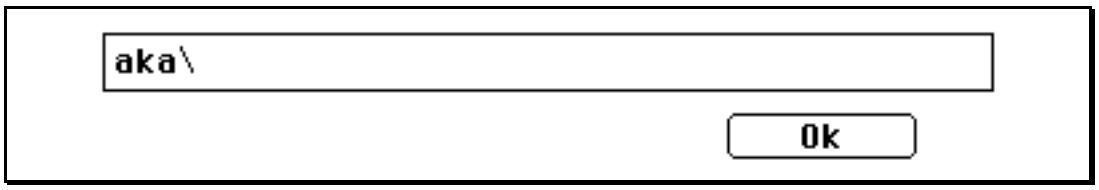
16-bit signals have a minimum value of -32766 and a maximum value of 32766. Their mean value tends to be around 0.

The minima and maxima of 8- and 12-bit signals are offset towards 0 by the mean value if you set "Zero Baseline" in File>File Setup.

Changing Title, Amplitudes and Sampling Frequency

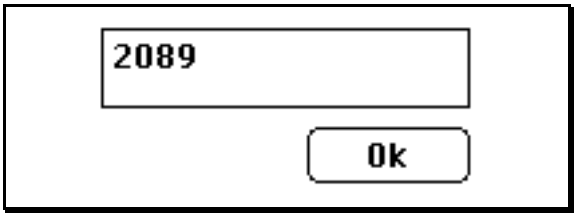
The signal's name, sampling frequency and individual sample amplitudes can be changed.

- Select the Signal Window.
- SHIFT-click on the signal title in the Signal Control Panel. A small popout dialog box opens.
- Type in the new name and click on "Ok". The name can be as long as 64 characters, but only the first 10 or so characters are shown in the title box on the control panel.
- To see the full name, click on the title to transfer it to the Edit Window. Alternatively, re-enter the edit window with SHIFT-click.



Change the *amplitude* or *sample value* of individual samples or the *sampling frequency*.

- SHIFT-click on the sample amplitude (bottom number) or on the sampling frequency. The popout dialog appears.
- If the shown value is changed, the signal's minimum and maximum values are recalculated, and the signal is replotted with the new values.



To Align or Time-Offset Signals
- Activate Signal>Set New Start (or COMMAND-comma).
- Wait for the instruction box to go away.
- Click in the place where the signal should start.
- A corresponding time offset is inserted in front of the signal.
- To return the signal to its original zero-offset condition, activate the menu command Signal>Original Start.

Signal Editing

Signalyze gives you a number of special tools to edit your signal.

You can—
Cut out or copy segments of a signal
Paste segments elsewhere
Smooth the transitions across the splice

Edit Buttons
Which kind of editing you can do depends on which *edit mode* the program is in. This is shown by the Edit Buttons:

Signal Editing Mode
Text Editing Mode
One-shot Transfer Mode

Switch between signal and text editing modes by clicking on the Edit Buttons or by clicking in the appropriate edit area (Edit Window or signal tracks).

The one-shot transfer mode can only be entered by clicking on the Edit Buttons. If selected, “Edit>Copy” causes the entire contents of the Edit Window to transferred to Clipboard, from where it can be inserted into another program. This way, a whole series of TAB-delimited numbers can be transferred into another application in “one shot”. This edit mode is not sensitive to clicks, and stays active until another edit mode is selected.

Cut, Copy, Paste, Clear
These four classic commands work with signals just as they would in text editing. The insertion point is placed *after* the newly pasted signal segment, and the signal display is scrolled to keep the insertion point visible.

Signal Transition Smoothing
Signalyze provides two options to smooth a signal splice.

Transition smoothing
After a signal editing operation that leaves or creates two edges of a signal, transition smoothing splices together the two edges. E.g., in moving a signal segment with "cut" and "paste", a first transition is created where the segment is removed, and two more transitions are created where the segment is inserted.

Signalyze's transition smoothing option (selected through Edit>Signal Edit) establishes a straight line between the xth sample on either side of the transition. All intervening samples are set to intermediate values between the two samples. X is set with "samples". Values between 3 and 20 often give reasonable results.

Tapering
Even signals smoothed with an averaged transition tend to retain a distinct click when reproduced. A better transition results from tapering, where a cosine function slowly reduces signal values to baseline over the specified duration. Very short tapers (e.g., 1-3 ms) are used to splice similar audio material, longer tapers (20-50 ms) serve to splice dissimilar audio materials.

Operation for simple splice after "cut" or "clear":

— Specify taper length in Edit>Signal Edit
— Perform Edit>Cut or Edit>Clear
— Execute Edit>Taper Left and, without changing cursor position, Edit>Taper Right

Suggestions
You're likely to get the best transitions by observing the following three rules:

- Make sure the two splices are at the signal's resting (i.e., average) level,
- In a section with regular oscillations, try to maintain the waves' regular form,
- Don't apply transition smoothing to too many samples (usually not more than 10).

Automatic trimming

Some signals have long silent periods at the beginning and the end. These can be trimmed off rapidly by the techniques available through Transform>AutoEdit Setup. In this dialog, the “threshold” establishes which noise level is considered part of the “silent portion”, to be distinguished from significant speech sound. The “margin” specifies how much silence is to be left before and after the significant speech sound after trimming.

Tutorial 7: Transformations, amplitude measurement and filtering

In this chapter, a short introduction is given to some of the most common transformations, as well as to amplitude measurement and signal filtering. All transformations are found in the "Transform" menu. Each type of transformation has a setup dialog and below it, a corresponding execution command.

Simple Transformations

By means of arithmetic and transcendental transformations, a few handy signal operations are possible. Here's a sampling:

Adding Noise to a Signal

- Obtain a source for continuous white noise, e.g., tune an FM radio to an empty frequency band.
- Make a first signal with white noise, but keep the input volume relatively low. Store the signal.
- Record and store a second signal with speech, also with relatively low input volume.
- Mark the first signal with the **1** - button.
- Mark the second signal with the **2** - button.
- In Transform>Simple Transformation Setup, specify "add" and "second signal". Exit with "Do". The two signals are added together.
- If the combined signal exceeds the 16-bit limits (shows "clipping" or other disturbances), reduce the bit width of one or both input signals with the Complex Transformation in Transform>Complex Transformation (e.g., reduce the bit width from 16 to 14, see below for details).

Enhancing the Noise or Echo Component of a Signal

- A signal's noise or echo component is typically characterized by small numeric values. Transforming the signal with a square root transformation increases the relative strength of small numeric values and thus enhances a signal's noise or echo component.
- Select a signal.
- Select "sqrt" in Transform>Simple Transformation Setup.

- Exit with "Do".
- Try raising a signal to power 1.2 for *reducing* a signal's noise component. You may have to down-shift a signal to 12-bit width first.

Complex Transformations

Signalyze's Super-Transformation:
Change or Leave Intact Any Of The Following Parameters:
Sampling Frequency, Bit width, Offset Bias, Numeric Type

The Complex Transformation (in Transform>Complex Transformations) modifies signals according to any of four parameters: sampling frequency, bit width, bias and numeric type. Any of these can be modified and any can be held stable.

CAUTION: Some of these modifications are irreversible. Make a backup of your signal beforehand.

Here's how the various modifications are performed:

- *Sampling frequency:* Modified by averaging or interpolating samples at selected intervals. Examples: A 11 kHz signal is converted into a 10 kHz signal by averaging every 10th and 11th sample, storing the average in the place of the 10th sample and removing the 11th sample.

A 10 kHz signal is converted into a 11 kHz signal by interpolating between every 10th and 11th sample and storing the interpolated value after the 10th sample. For greatest speed, the calculation of which samples to average or interpolate from is performed in fixed-point math.

- *Bit width:* Modified by left- and right-shift operations. Example: A 16-bit signal is converted into an 8-bit signal by right-shifting each sample by 8 binary positions. Or an 8-bit signal is converted into a 12-bit signal by left-shifting each sample by 4 binary positions. The new binary positions created to the left or right of the number are filled with zeroes.

By the way, a signal's bit width is calculated in the following manner: (1) Find the signal's maximum and minimum. (2) Determine which is greater when taken as an absolute value. (3) Double the greater of the two. (4) Determine the bid width that will cover the obtained range.

- *Bias (or "offset bias "):* Modified by adding the specified number to the signal's linear average.
- *Numeric type:* Sample values can be either entirely in the positive domain (between 0 and 32,767) or can extend over both, positive and negative domains (between -32,767 and +32,767). Different digitizing devices use different numeric types, so your signals may be natively "unsigned" (entirely positive) or "signed" (positive and negative, also

called "complements-two"). When the numeric type is changed with the menu popup, an appropriate offset is suggested in the offset bias edit box. For signed output signals, the offset is 0. For unsigned output signals, the offset depends on the bit width. In the case of bit widths 8-15, it is 1/2 of total range. In the case of bit width 16 it is set to 0, due to the range limits of a 16-bit signal.

By means of this transformation, you can transform nearly any signal format into nearly any other. In fact, *this transformation is applied automatically to save signals to file in formats that require specific sampling rates, bit widths and offset biases,* so there's no need to perform such a transformation manually prior to saving the signal in a non-Signalyze format.

Enhance the High Frequency Components of a Signal

Differentiation is a fast way of enhancing the high frequency components of a signal. Differentiation proceeds by taking differences between succeeding samples. E.g.,

ORIGINAL SIGNAL	DERIVATIVE
400	
300	-100
50	- 250
75	25

- Select a signal.
- Select "Derivative Differences" in Transform>Complex Transformation Setup.
- Exit with "Do".

Amplitude Envelopes

Measure Amplitude with an RMS Envelope

Envelopes are used to investigate the overall amplitude development in speech. Here we'll illustrate the RMS envelope which permits the taking of dB measurements.

The RMS envelope routine first calculates a *power envelope* by entering the squared value of each sample of a given window (specified in ms in Transform>Envelope Setup) into the window sum, and by taking averages over the window. Squaring the values moves negative samples into the positive domain. The RMS envelope is calculated from the power envelope by taking the square root of each mean. Hence the name: "RMS" or "root mean square", i.e., "the root of the mean of the squares in a given window". The output of an RMS is thus akin to a

mean value of the rectified signal values, taken over the width of a window.

- Select a signal.
- Select "RMS Envelope", "50 ms" and "Raw values" in Transform>Envelope Setup.
- Exit with "Do".

The RMS values are drawn below the signal.

Calculate the RMS as dB differences to the value 1

Amplitudes as they are usually measured in speech research are rather imprecise entities. Generally, recording is performed in more or less noisy and reverberating environments, with little control over the distance between the speaker and the microphone. Consequently, no inference is possible concerning the *absolute* amplitude of the recorded sound. Moreover, microphones and recording devices are not entirely linear, i.e., some frequencies are favored over others. Finally, amplitude is limited by the signal's bit width, severely so when recording at 8 bits.

For all these reasons, it is wise to approach amplitude evaluation with prudence. The safest measure involves the comparison of two sounds of a certain amplitude (e.g., the RMS of two vowels, see next section). Here, the signal is generally of sufficient strength to provide a reasonable estimate of amplitude differences between the two sounds, at least as they are perceived in reproduction by a listener.

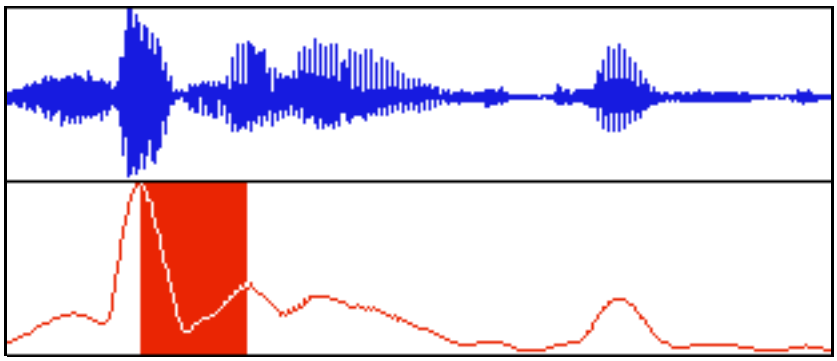
Amplitudes can also be compared to the smallest possible amplitude which is 1. This is done when the option "dB differences from 1" is selected in the Envelope Setup dialog. The curve obtained in this way is based on a comparison of each amplitude to the value 1. In this type of curve, smaller amplitudes are evaluated less precisely than greater amplitudes.

- Select a signal.
- Select "RMS Envelope", "50 ms" and "dB differences from 1" in Transform>Envelope Setup.
- Exit with "Do".

The dB values of the RMS are given in "times 100" format, e.g., a dB difference of 34.56 is given as 3456.

Measure Relative dB Differences in an RMS Envelope

- In Edit>Show Statistics Options, select "dB difference between left and right end point values". Exit with "Save".
- In an RMS (or power) envelope, blacken the distance between two vowels, as shown above.
- Do "Edit>Show Statistics" or COMMAND-U. The dB-difference between the two vowels is displayed.



Raw RMS values are converted to dB by means of the routine:

```
if (min < 1) min = 1;
pre_dB = RMSvalue /min;
dB = 20 * log10(pre_dB);
```

CAUTION 1: The speech amplitudes encoded by 8 bit signals are necessarily limited. Compare, for example, the maximum dB differences encoded by an 8-bit and a 16-bit signal:

maximum difference in an 8-bit signal: 127

42 dB = log10(127/1) * 20

maximum difference in a Signalyze 16-bit signal: 32765

90 dB = log10(32765/1) * 20

So if you work with 8-bit signals don't be surprised if your measures give you different results from those you find printed in the literature. Published work on amplitudes is generally based on the analysis of 16-bit (or even higher-bandwidth) signals.

CAUTION 2: Since RMS values give you an *average* which generally stays well below the peak signal values, it is not surprising to find amplitude difference for RMSs that diverge quite a bit from those measured for isolated peak values.

CAUTION 3: The precision of dB measurement is very sensitive to the reference minimum. Example:

Measured RMS amplitude in vowel: 10000
Measured RMS amplitude in quiet portion A: 5
Measured RMS amplitude in quiet portion B: 1

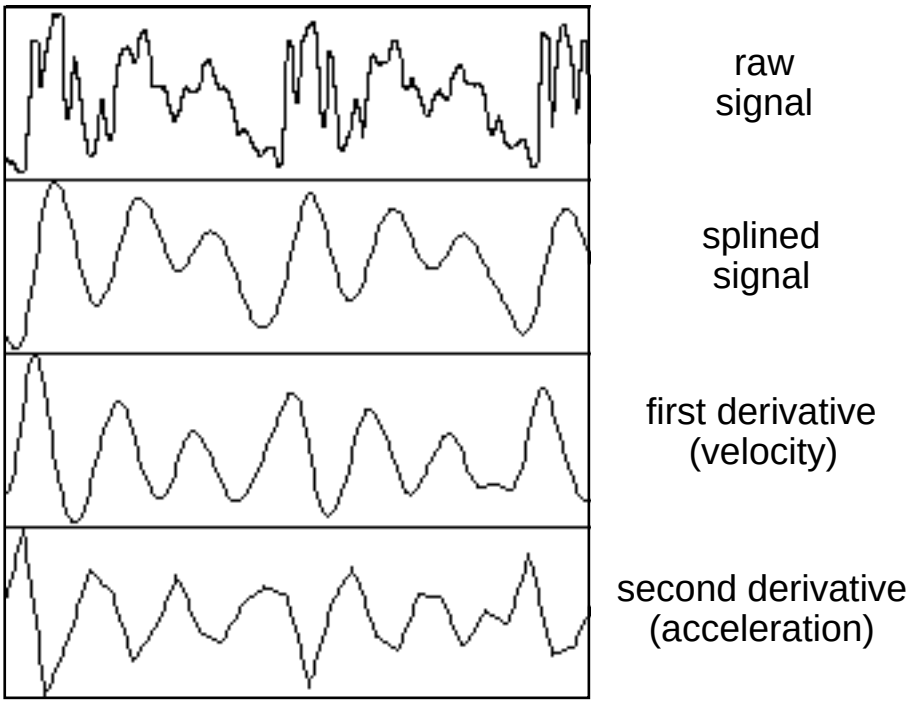
dB difference vowel-portion A: 66dB=log10(10000/5) * 20
dB difference vowel-portion B: 80dB=log10(10000/1) * 20

In other words, the 14 dB difference between the two measures is due to a minor "bump" in the reference value in the signal's quiet portion. Consequence: The most reliable measures are those made from peak to peak of relatively strong portions of the signal.

Save dB Differences Relative to the Minimum Value
When the result of a power or RMS transformation is saved in TEXT format (File>Save As), the user is given an option to save the result in dB format (with decimal points). The dB values are calculated relative to the value 1.

Splines: A Set of Routines for Kinematic (Movement) Signals

General Information
Splines are part of a family of curve fitting techniques that smooth a signal while attempting to remain true to observed values. Splines do so by means of defining polynomial functions, which permits the easy extraction of a curve's first and second derivatives. This makes splines particularly useful for kinematic (movement) signals, since the first derivative corresponds to velocity and the second derivative to acceleration. We'll illustrate the use of the average spline here.



First and Second Derivatives by an Average Spline
- The average spline first calculates adjoining averages (so-called "knots") and then connects the knots by curves which are described as polynomial functions. Minimum: 4 knots.
- The "Knot width" corresponds to the width of the sampling window over which averages are calculated. With greater knot widths, the resulting curve is smoother, but further removed from the observed

values. With smaller knot widths, it is rougher, yet closer to the observed values. The researcher must determine the knot width that represents a reasonable compromise between an approximation to the data and an optimal smoothing of the curve.

- Splines require a great deal of calculation memory. To illustrate spline operation, select (blacken) a *small* portion of any signal.
- Make sure the signal is maximally 12 bits wide (check Transform>Complex Transformation Setup). If it's a signal with greater bit width, reduce the signal to 12-bit width by means of the "Complex Transformation".
- Make sure there are three empty tracks below the signal.
- In Transform>Spline Setup, choose "Average Spline" with a knot width of 10. Select all three options: "Splined Signal", "1st derivative" and "2nd derivative". Deselect "Observe".
- Exit with "Do". The routine first calculates the averages and then the splines.

Zero Crossings

General Information

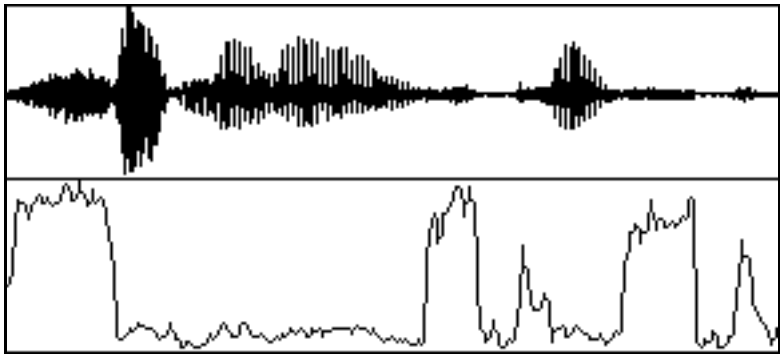
The frequency of zero crossings is a widely-used index of frication in speech analysis. The highly irregular signals found in [f], [s], [ʃ] and other fricatives tend to cross the signal's resting value much more frequently than, for example, the regularly oscillating signal found in vowel portions of the speech signal. In Signalyze, the "resting value" is given as the linear mean of all samples.

The width of the analysis window is given in ms. E.g., with a 10-ms window specification, zero-crossings are counted over successive 10-ms windows. A new window is calculated every sample step. The resulting count is multiplied by 100 to obtain crossings per second. This frequency is displayed for the width of the analysis window.

CAUTION: Strongly voiced portions of speech carrying frication may be missed by the zero crossing routine, since the voice oscillations deviate considerably from the signal's mean value. Zero crossings may thus be taken as a first index of frication, but should not be considered to be definitive arbiters concerning the presence or absence of frication.

Calculate a Frication Index With Zero Crossings

- Select a signal.
- In Transform>Zero Crossing Setup, specify a window width of 10 ms.
- Exit with "Do". Frequency of zero crossing is given in Hz.



Filtering

General Information

Filtering is common in signal analysis, but it is not a well-known topic in the speech sciences. To cover this gap to some extent, a short introduction to filtering is given in the following section.

What's a Filter and When to Use It

- *Definition.* A filter changes the frequency characteristics of a signal. Typically, certain frequency components in the input signal are suppressed while other frequency components are preserved.
- *Analog vs. digital filters.* Both analog and digital filters are used in signal analysis. An analog filter consists of an electric circuit. A digital filter performs its filtering functions by computational means.

The major advantage of an analog filter is its real-time operation: Signals are filtered as fast as they are recorded. On the other hand, good analog filters are expensive and their effects are irreversible: Once a signal is filtered, the filtering effects cannot be undone. The major advantage of a digital filter is its flexibility: A signal can be filtered in different ways to suit various purposes. Among its disadvantages is its slowness of operation, at least on the more common Macintoshes. (A floating-point 5th order Butterworth filter is essentially real-time on a Quadra).

Consequently, analog filters are often used for clearly identified purposes, e.g., in anti-aliasing filtering (see next paragraph). Digital filters are favored in cases where several filtered versions of a signal are required, as in various forms of signal processing.

When to Use Filtering.

— *Anti-aliasing filtering.* Signals should *always* be filtered prior to spectral analysis ("anti-aliasing filtering"). In this type of filtering, high-frequency components of a signal are strongly reduced, leaving only low-frequency components "eligible" for spectral analysis ("low-pass

filtering"). If high-frequency components were not suppressed, they would periodically reinforce low-frequency waves, leading to erroneous spectral analysis results.

PLEASE NOTE: Most current A/D devices incorporate an analog anti-aliasing filter, so the user need not be concerned with this issue. Other devices incorporate some analog filtering, but the high-frequency attenuation is insufficient for research purposes.

To find out if input device anti-aliasing is sufficient, perform spectra on a number of clearly-recorded vowels before and after low-pass filtering. If there is no significant difference in the appearance of the spectrum below the cutoff frequency, the analog filtering is satisfactory. The cutoff frequency should be set at somewhat below half of sampling frequency (e.g., at 40%), below the so-called "Nyquist frequency" (which is 50% of sampling frequency). Example: for a 20 kHz signal, the cutoff frequency should be at 8 kHz to make sure that the complete range of frequencies below 10 kHz is protected against aliasing effects.

— *Signal Processing.* Filtering is also commonly used for signal processing, as in the preparation of stimuli in psycholinguistic experiments or to "clean up" difficult-to-perceive signals.

Finite vs. Infinite Impulse Response Filters (FIR vs. IIR Filters)

There are two major ways of building digital filters. They are distinguished by whether a given pulse in the original signal is reproduced as a finite or infinite pulse in the filtered signal.

• *FIR filters*

To illustrate the operation of a FIR filter, imagine a first signal in which a group of samples all have the same high value, while the rest of the samples are at zero. Graphically this signal looks like a square pulse. Now assume that you're building a second signal by taking a set of averages over a given "window" of samples in the first signal; further assume that this window is wider than the pulse. A second signal can be built from all the averages, one for each window position, as the window is moved from left to right.

As the window's right edge reaches the pulse's left "wall", the averaged output starts rising from zero. The window's output keeps increasing as more of the pulse's values are entered into the average. After the window's right edge leaves the pulse's right wall, the average values remain stationary, until they start decreasing when the window's left edge goes past the pulse's left wall. Finally, the average value drops back to zero. The output of this filtering operation is a second, smoothed pulse, resulting from the finite averaging response of the sliding window. Because of the averaging, the new signal begins half a window later than the original signal. Once this time offset is subtracted, the new pulse's

center is found at the same place as the old pulse's center. As the pulse is of finite duration, this operation is known as the application of a "finite impulse response (FIR) filter".

It is noted that the output signal "looks essentially" like the original signal. In other words, an FIR filter can be written in a way so as not change the phase characteristics of the original signal. This is in difference to the Infinite Impulse Response (IIR) filters discussed below.

FIR filters are described by the form of their averaging window. Moving average filters have a square response, i.e., each sample counts equally strong in the average. Other FIR filter windows have forms ("distributions") designed to reduce secondary deformations of the signal, such as triangular, gaussian ("normal") or Hamming distributions. In these windows, samples at the center of the window contribute more strongly to the average than samples close to the window's two edges. Even on small computers, FIR filters can be written to operate quite rapidly.

• *IIR filters*

The other classic approach towards filtering involves modifying each sample "as it comes along" in the input stream. The modification to be applied is calculated ahead of time in the form of "filtering coefficients".

It is possible to apply a single modification ("filter order 1") or multiple, sequentially applied modifications ("filter orders 2 and higher"). The effect of these modifications is (at least in theory) infinite, i.e., a given impulse "propagates its effects" upon a theoretically infinite number of subsequent samples in the filtered signal.

This "propagation effect" is greater with higher-order filters than with lower-order filters. Consequently, the center of an output pulse is right-displaced with respect to the center of the input pulse, and at higher filter orders it is more strongly displaced than at lower filter orders. At the same time it can be seen that this "delay" generally causes a fairly strong deformation of the phase characteristics of the signal, as compared to the original signal.

This approach to filtering is very flexible, since it is possible to define sets of coefficients that simulate all major types of filters: high-pass (suppressing low frequencies), low-pass (suppressing high frequencies), band-pass (suppressing both high and low frequencies) and band-clip (suppressing energy between two frequencies). Two well-known IIR filters are the Butterworth and Chebychev filters. IIR filters tend to be a fair bit slower than FIR filters, especially on less powerful computers.

A Simple FIR Filter: A Moving Average

- Select a signal.
- In Transform>Filter Setup, under "Type", select "Moving Average".
- Specify a window size of 10 ms.
- Exit with "Do". A filtered signal is created.
- Experiment with different window sizes. At very narrow window widths, the abruptness of entering new values at their full strength may introduce undesirable side effects. At very wide window widths, the essential features of the signal may be lost.

A Standard IIR Filter: A Tenth-Order Butterworth Low Pass

- Select a signal.
- In Transform>Filter Setup, under "Type", select "Butterworth".
- Specify "Low Pass" filtering and set the "High" cutoff at 2000 Hz.
- Set the order to 10.
- Deselect "Floating Point".
- Exit with "Do".
- A filtered signal is created.
- Make narrow-band spectra of both the original and the filtered signal. It will be found that the lower frequencies are intact and that frequencies above 2000 Hz are strongly suppressed.
- Note on use of floating point:

  — On the more powerful Macintoshes equipped with co-processor, it is a good idea to leave the floating point option **on** at all times. The time loss is minimal, and the user is always assured of optimal precision.

  — On the weaker Macintoshes, much time can be gained by deselecting the floating point option. However, the user must be on the watchout for possible overflow conditions, particularly with high order numbers and low frequency cutoffs. In those cases, the signal tends to look very deformed. If overflow conditions are encountered, the floating point option should be selected.

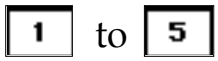
Tutorial 8: Advanced Operation

In this chapter, a number of advanced topics are covered: Menu command playback keys, file formats, printing, saving signal and spectrogram images, setting and controlling memory, working in different languages, batch loading, and sequential macros.

Record Menu Commands

The row of buttons at the top of the General Control Panel is used to store often-used menu commands. Their operation is nearly self-explanatory.

Record Menu Command


Play back Menu Command

- Suppose you often use the menu command Transform>Simple Transformation.
- Select a signal.
- Click on the Record Menu Command button ( ).
- You get a message telling you to select a playback button (to).
- Click, say, on button number 2.
- Now you get a message asking you to select a menu command. Go to the menu bar and click. Wait till the message disappears and then select the command Transform>Simple Transformation.
- From that moment on, clicking on playback button number 2 causes the stored menu command to be executed.
- The button command is valid till you record another command into the same button.
- For technical reasons, the menu items under the Apple Menu are not eligible for macro storing.
- If you've forgotten which menu command is stored in a given button, click on it while holding the SHIFT or the OPTION key. Two lines are shown: At top is the menu and below it is the menu item. If nothing is stored, an error message is generated.

Sequential Macros

Signalyze contains a simple sequential macro facility that lets you run through a series of processes and transformations.

Build and Call a Sequence of Macro Processes

- In Macro>Sequential Macro Setup, select processes in the desired sequence. The sequence number appears to the left of the process. Select again to eliminate a process from the sequence.
- The bottom five entries in the Sequential Macro Setup represent the recordable menu commands. By first recording menu commands into the various buttons and by chaining these buttons into a sequential macro, you can program nearly any small chain of operations possible in Signalyze.
- To execute the macro, select the menu command Macro>Play Sequential Macro (COMMAND-4). All processes apply to the currently selected signal, and are performed as defined in the various setup dialogs.

Example: Interactive Pitch Extraction

If you have a sound digitizer that can be connected to Signalyze, you can try the following handy application:

- Enable and set up the digitizer through the Macro>Record Setup, as described in Tutorial 3.
- For a start, specify 2 seconds recording time. If there isn't 2 seconds available, add a few tens of kilobytes to signal memory via Edit>Startup Options, quit the program and return to this dialog.
- Quit with "Save".
- Put the menu commands "Macro>Record Signal" and "Signal>Delete All Signals" into the Menu Command Playback Buttons 1 and 2.
- Open Macro>Sequential Macro Setup. Select (1) "Btn 1: Macro>Record Signal" (this calls what you stored in the Menu Playback Button 1), (2) "Play Audio" and (3) "Pitch extraction".
- Quit the Sequential Macro Setup with "Save".
- Set pitch extraction parameters for your own voice with Spectral>Pitch Extraction Setup. Use the routine that works best for you (see Tutorial 5). Quit with "Save".
- If necessary, disable the cautionary "Save Signal?" dialogs by disabling the menu command Signal>Delete Save Dialogs. (The hook mark disappears).
- Start recording by calling "Play Sequential Macro" (COMMAND-4). Start talking when the cross cursor appears in the dialog, and stop

talking just before it disappears (you'll get the feel for the time lapse with a few tries).
- After the recording is made, the remaining Sequential Macros take over. You get an automatic playback of the recording, followed by a pitch extraction.
- Erase both signals with Menu Playback Button 2 and start over with "Play Sequential Macro".

In this manner, Signalyze can act as an interactive pitch extractor for use in foreign language teaching and speech therapy sessions.

File Formats

In this section, signal files, text files, graphics files and signal file conversion techniques are discussed.

Signal File Formats

Sound signals on the Macintosh come in a bewildering variety. In this section, the main characteristics of different signal file formats are explained.

At the present, no single sound file format has gained universal status, either inside or outside of the Macintosh world. Users that employ a number of different programs or computers for their signal processing must necessarily juggle between a number of different signal file formats.

Signalyze tries to make the juggling as painless as possible. The most common Macintosh formats are fully incorporated into the program. Moreover, the program "FileConverter" (written by Stefan Werner and supplied free with the program) can convert a number of additional file formats off-line. Finally, an open invitation is extended to all Signalyze customers to send us the specifications (or simply some sample files) for signal file formats which we do not presently support. If a file format is in reasonably common use, either on the Mac or in another environment, we'll do our best to provide a translator for it (see box).

As you juggle between signal file formats, you may wonder which format to adopt for most of your work. If you're planning to do much work with Signalyze, the two formats "Signalyze" and "Audio IFF" (AIFF) can be recommended for the following reasons:
- They load into the program faster than most other files.
- They can encode nearly any sampling frequency. Most other file formats lock you into a very reduced choice of sampling frequencies.
- They are the only files that can be loaded into Signalyze by double-clicking or in batch load (see below).
- They will remain compatible with future versions of Signalyze.

- The saving of other formats may involve a normalization to the target format's numeric type, bit width and sampling frequency. This often changes the actual signal values and frequently involves a loss of signal precision. If many such normalizations are performed on a signal, they can end up inducing serious signal degradation. Saving in AIFF format never changes the signal values or signal precision. The same is true for Signalyze format, except that sampling rate is precise only to the nearest kilohertz in the case of sampling rates above 32 kHz.
- The Signalyze format is exceedingly simple (see the "Technical Information" section). If you ever need to write your own routine to read Signalyze binary files, it won't take a Ph.D. in computer science to do so.
- The Signalyze environment provides back-and-forth translators for most other file formats, which is not the case of any other file format we're familiar with.

If you have a file format for which you need a translator:
- Record three or four *speech* signals (not music) at different sampling frequencies onto a diskette.
- If you happen to have a description of the file format, make a copy of it.
- Provide a short indication of the name of the file format, the program that uses it, and the computer(s) it is used on.
- Send the whole bundle to InfoSignal Inc.
- We'll get back to you shortly.

8- and 16-Bit Signal Formats

You won't hardly notice as Signalyze switches back and forth between 8-, 12-, and 16-bit files format. Still, some differences between these formats are important to know about. So here's a summary of the various strengths and weaknesses of these formats:

8-bit Format

Where used: The original Macintosh sound format, used in standard sounds by the Macintosh operating system (beeps etc.). Also used by HyperCard, the 8-bit Apple Sound Input Device, MacRecorder, and many other applications and devices.

Where used in Signalyze: For the built-in input and output audio on machines that only support 8-bit sound I/O (all machines prior to the Macintosh AV and the Power Macintosh series).

How the data portion of the file looks: An 8-bit file is a long string of ASCII characters. Each encodes one of 256 possible voltage levels.

How 8-bit signals are represented in Signalyze: As 16-bit integers, where only a small range of the full 16-bit range is used.

Advantages: Commonly used format, makes files that are only half as long as comparable 12- or 16-bit files.

Disadvantages: Gives sound that is distinctly inferior in quality due to small dynamic range, difficult to record well, particularly with voices that vary a considerably in amplitude (e.g., children, speech-impaired patients), limited spectrographic quality in the higher frequency ranges, unreliable amplitude information.

12-bit Format

Where used: On many traditional speech analysis systems, 12-bit signals were standard. For a period on the Mac also, the only high-quality sound files were the MacAdios 12-bit files.

Where used in Signalyze: To guard against 16-bit overload, a fair number of Signalyze processes operate on 12-bit versions of the signals (e.g., FFT-based spectral analyses, splines, correlations). Prior to the application of such processes, copied portions of the signals are bit-shifted into 12-bit status, but the original signal's bit width is never changed.

How the data portion of the file looks: An 12-bit file is a long string of 16-bit integers (samples). Each encodes one of 4096 possible voltage levels.

How 16-bit signals are represented in Signalyze: As 16-bit integers, where only a small range of the full 16-bit range is used.

Advantages: Provides sound that is better in quality and signals whose high frequency spectral characteristics are distinctly enhanced, in comparison with 8-bit signals. 12-bit signals also encode greatly varying amplitudes much better than 8-bit signals.

Disadvantages: 12-bit signals leave 4 bits unused in each 16-bit integer, which is wasteful of disk space (unless you compress the file). Also, research on high-quality acoustic applications in speech, music and animal sounds often depend on first-class sound recording and sound reproduction which is not possible with 12-bit signals.

16-bit Format

Where used: Most modern DSP ("digital signal processing") and A/D-D/A devices use 16-bit formats.

Where used in Signalyze: All Signalyze processing operates by default on 16-bit format.

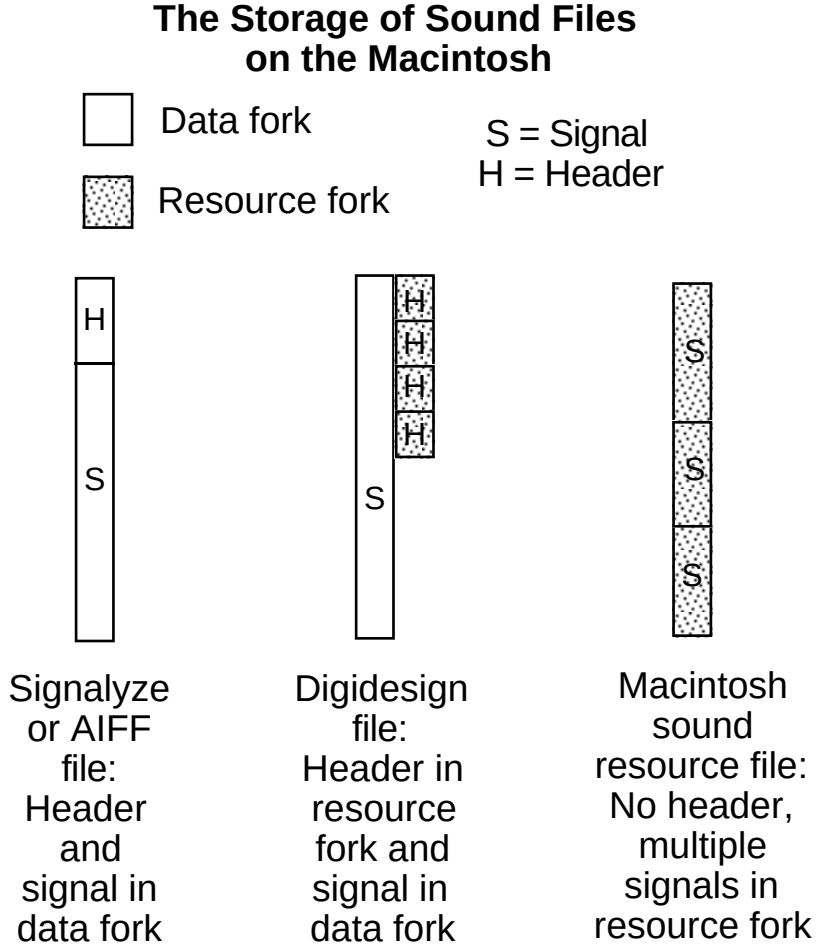
How the data portion of the file looks: An 16-bit file is a long string of 16-bit integers. Each encodes one of 65,536 possible voltage levels.

How 16-bit signals are represented in Signalyze: As 16-bit integers.

Advantages: If processed by high-quality input/output-devices, 16-bit signals provide outstanding sound quality. Such signals have

excellent high frequency spectral characteristics. Greatly varying amplitudes pose practically no encoding problems in a 16-bit environment.

Disadvantages: 16-bit signal files take a fair bit of space on disk and permit little compression (depending on the algorithm, 0-10% compression possible).



Signals Stored as Files or as Resources

On the Macintosh, there are two ways to store material in a file:

- In the data fork of a file
- In the resource fork of a file

Different formats use the data and resource forks in different ways. Here are some examples:

- Signalyze, AIFF and MacADIOS format use only the data fork of a file. They reserve an initial portion of the file (a "header") to store the essential information about the file (e.g., sampling frequency).
- Digidesign (AudioMedia, Sound Designer II), SoundEdit and SoundWave use a mixed storage pattern. The signal information is stored in the data fork and the header information is stored as a set of resources.

- Macintosh sound (type 'snd ') resources are capable of storing one or several signals in the resource fork.

This information becomes important to the user on the following occasions:

- Loading a longish *sound resource* into Signalyze easily provokes the "out-of-memory" message. This is because the entire resource must be loaded at once into calculation memory, from where it can be translated into 16-bit format and moved to signal memory. Increase calculation memory through Edit>Startup Options if you encounter this message.
- Moving a signal file to another computer is less complicated if all the information is in the data fork, than when some or all of it is in the resource fork. Signals that contain everything in the data fork can be sent directly to the other computer, where they can be decoded. If information is found in the resource fork, it has to be decoded first and converted to data fork information, since other computers do not generally maintain a resource fork.

Remember:

- You may lose bit precision and sampling frequency precision by saving a file in non-Signalyze format.
- ASCII-format, AIFF and Signalyze saves never cause any loss in bit precision or sampling frequency (Signalyze: if sampling frequency is under 32 kHz).

Signal File Input and Output

- The *inputting* of non-Signalyze format files is totally transparent to the user. All format conversions are handled automatically.
- **IMPORTANT:** The *outputting* of non-Signalyze format files involves a *normalization*. That is, the signal is modified for bit width, numeric type, bias offset and sampling frequency, to render it similar to a native file of that format. If many such normalizations are performed on a signal, they can end up inducing serious signal degradation.
- You can define the percent of acceptable modification of the sampling frequency through File>File Setup *("Output tolerance")*. An output tolerance of 50% means that you're willing to accept up to a 50% up- or down-sampling change as you save a signal in a non-Signalyze format.

Here are some examples (output tolerance = 50%):

	sampling frequency (Hz)	**bit width**	**offset bias**
Signalyze signal 1:	*16,000*	*16*	*0*
saved as MacAdios file:	20,833 (upsampled)	12 (right shift 4 bits)	2048 (12-bit unsigned)
saved as SoundEdit file:	11,128 (downsampled)	8 (right shift 8 bits)	128 (8-bit unsigned)
saved as AIFF file:	16,000 (no change)	16 (no change)	0 (no change)
saved as Signalyze file:	16,000 (no change)	16 (no change)	0 (no change)
saved as ASCII file:	16,000 (no change)	16 (no change)	0 (no change)
Signalyze signal 2:	*11,128*	*8*	*128*
saved as MacAdios file:	10,416 (downsampled)	12 (left shift 4 bits)	2048 (12-bit unsigned)
saved as Digidesign file:	8,000 (downsampled)	8 (left shift 8 bits)	0 (16-bit signed)
saved as AIFF file:	11,128 (no change)	8 (no change)	128 (no change)
saved as Signalyze file:	11,128 (no change)	8 (no change)	128 (no change)
saved as ASCII file:	11,128 (no change)	8 (no change)	128 (no change)

More on ASCII-Format Signals

ASCII format is a machine-independent format where each sample is given as a string of ASCII characters, separated by a specified ASCII character. Such files are marked as type "TEXT" in the Macintosh environment.

ASCII-Format signals come in two varieties:

- *With header:* The signal's title and sampling frequency are at the start of the file. That way they won't get lost when you transfer a ASCII signal file to another computer. However, most other programs that

accept ASCII-format signals go haywire when they encounter the header information.

- *Without a header:* You have to keep track of the signal's title and sampling frequency yourself. This type of ASCII-format signal can be read by a number of other programs (such as MATLAB).

> **RECOMMENDATION:** For transferring files to other computers, long and sometimes painful lab experience suggests that it is a good idea to use the header option. Too often, the operator loses track of which file is which and confuses file names. You are protected against this type of error if you use the header option.

ASCII-format Options In File>File Setup

- *Numeric Type in ASCII Format:* Data can be either whole numbers (integers) or decimal numbers (floating point). The decimal portion of a decimal number is discarded on input.
- *Separator:* Any single ASCII character can serve as separator between data points.
- *The special status of SPACE as a Separator:* The SPACE character (ASCII 32) is usually interpreted as a blank surrounding the number. However, you can also specify ASCII 32 as a separating character. In that case, it must be the ONLY separating character, otherwise the routine will be "confused".
- *Name and Sampling Rate:* With the header option, the signal name and the sampling frequency are furnished at the start of the file. If not, this information can be specified manually for each file through a special dialog that comes up before the file is read in, or it can be generated from the file's name and from the sample frequency specified in the File Setup dialog.
- *What Gets Saved:* Generally, it is the values making up the signal that get saved into the ASCII file. However, with spectra and with power- or RMS envelopes, a dialog pops up, asking if saving as dB-values is desired. If this option is selected, the signal's values are converted to dB values and are saved in the numeric format given in Signal>Numeric Setup.

Example of a ASCII-format Signal With Header

(¶ indicates separating character):

```
New Signal¶   - name
11200¶        - sampling rate in Hz
2045.0¶       - first and subsequent
2046.0¶       data points are given in
2047.0¶       decimal number format
```

```
etc.          in this example
```

Example of a ASCII-format Signal Without Header

```
2045¶         Data points are given in
2046¶         whole number format
2047¶         in this example
etc.
```

TEXT Files

Files opened and saved from the Text Window are standard Macintosh 8-bit TEXT files. That is, files that contain only characters of the full Macintosh ASCII set (including international characters), but no formatting information.

Visible Tabs (¬, see Edit>Show Values Options) are converted to and from "real" Tabs when the Text Window is saved to a file or when a new text file is read in.

Graphics Files

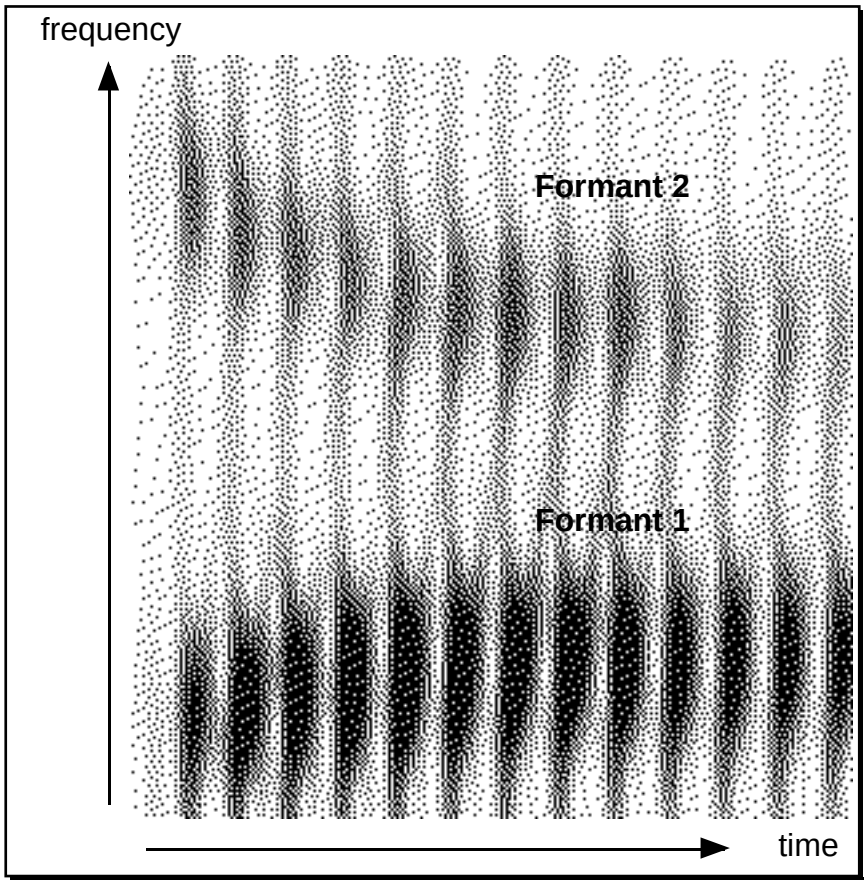
Files opened or saved through the commands File>Open NIH Image or File>Save NIH Image are in PICT format. This is a Macintosh-native format that can be read by a large number of graphics processing programs (Canvas, Superpaint, MacDraw, Photoshop, NIH Image, Claris Works, Microsoft Word, etc.).

Some interesting image processing is possible with spectrograms. The freeware program *NIH Image* (written for the NIH by Wayne Rasband and available from many electronic bulletin boards) can for example extract high-quality three-dimensional images from spectrograms. Also, it can provide high-quality dithering if you happen to have a non-PostScript printer (ink jet printer, etc.). However, you need a 256-color/grayscale monitor to use *NIH Image*.

Instructions for Generating a 3D Spectrogram through NIH Image:

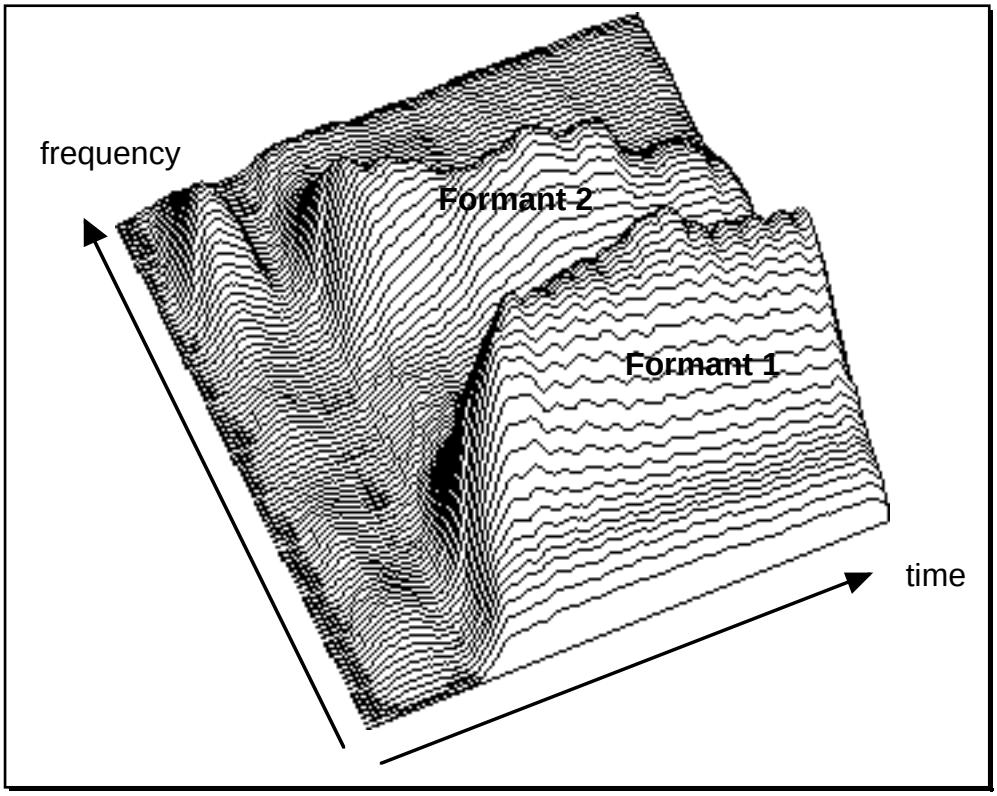
- Set the Monitors Control Panels accessory to 256 colors or grays.
- Create a gray scale spectrogram.
- Save it with File>Save Image.
- Load it into "NIH Image".
- Make a new window with "New".
- Copy and paste a selected portion of the spectrogram to the new window.

• Smooth this image by "Convolving" it with the file Gauss 15 x 15, found in the folder "Kernels". "Kernels" comes with the *NIH Image* program.
• Apply the menu command "Surface Plot".
• For instructive views of spectrographic features, rotate the grayscale spectrogram before applying the menu command "Surface Plot".

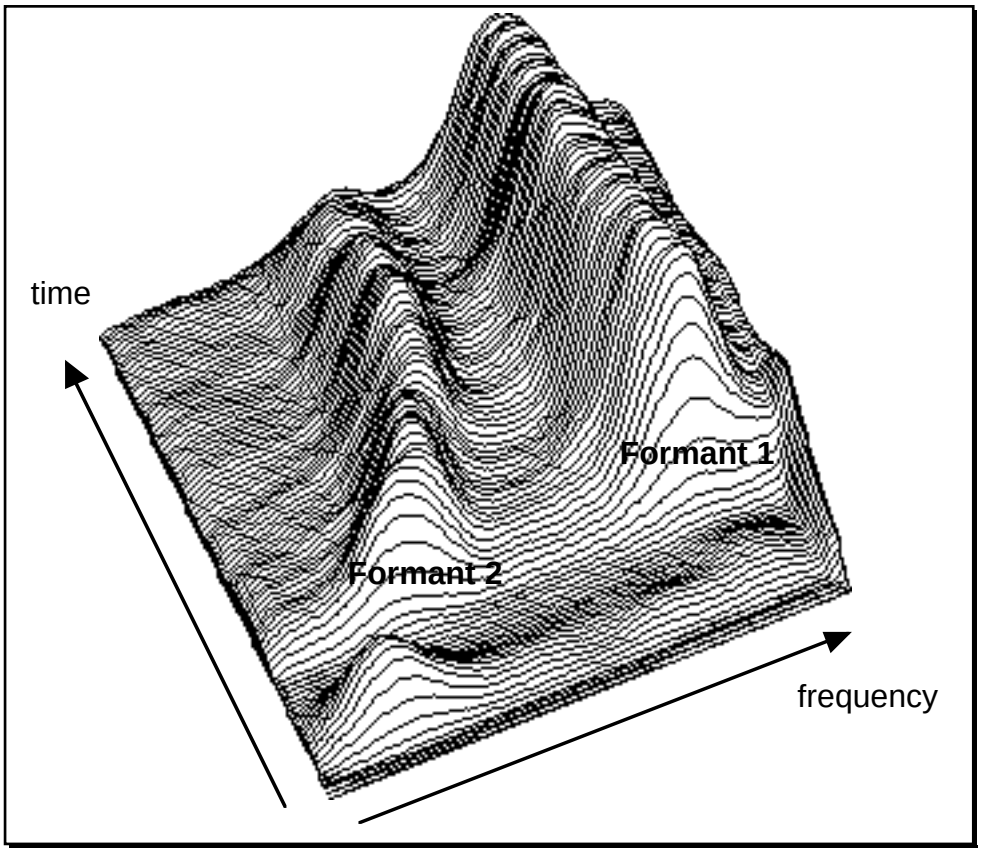


A 256-grays Signalyze spectrogram of the sound /ka/, dithered with the freeware program "NIH Image".

Instructions for Dithering a Spectrogram With a Graphics Program:
• Set the Monitors Control Panels accessory to 256 colors or grays.
• Draw a gray scale spectrogram on a separate window with "Plot every line". This creates an 8-bit image (each pixel occupies 8 bits).
• Save it with File>Save Image.
• Start up your image processing program, make sure it can read PICT format, and load the spectrogram image.
• If possible, scale up (enlarge) the image by a factor of two (200%).
• Dither the image, i.e., convert the 8-bit spectrogram into a 1-bit image. The darker colors become dense patterns of dots and the lighter colors will be sparser patterns of dots.
• If you were able to scale up the image, set 50% reduction in Page Setup.
• Print out the picture.
• Try scaling up 400% and printing at 25% for even better results.



Waterfall representation obtained with the freeware program "NIH Image" from the same spectrogram as used in the previous figure. Above: Side view. Below: Front view.



Printing

Signalyze permits the printing of
- Signal window
- Separate spectrogram windows
- Plot windows
- Text

Printer Settings
Signalyze uses Apple-native bit-image printing. Your printer must therefore be capable of printing such images. The ImageWriter, the LaserWriter or nearly any other Macintosh-compatible dot matrix, ink jet or laser printer is of this type. Text-only printers can't be used with Signalyze.

Printing is routed via the "Chooser" accessory in the -Menu. Make sure you select the right printer outlet (the "modem" or the "printer" port). If you've changed printer settings via the "Chooser" accessory, you must perform File>Page Setup after leaving the "Chooser" accessory, in order to let Signalyze know about the changes.

The Page Setup dialog also lets you choose a sideways (vertical) display, so that even larger signal or spectrogram displays fit on a single page. Further, you can request reduced-sized printing through the Page Setup dialog.

Printing a Signal Display With a Low-Resolution Spectrogram
The following illustrates black-and-white only printing.
- If you use a color or gray scale monitor, first select "Black/White" in the "Monitors" control panel.
- To print without reduction from a large screen, you may have to reduce the size of the Signal Window first. Change the startup values with Edit>Startup Options to "Manual Window Sizing", six displays per card, and set "Signal Display Width" to 554 pixels (as in an 11-inch screen). Exit the program and restart.
- If you're working with a large screen, select *sideways* display in File>Page Setup.
- If necessary, bring the Signal Window to the front.
- Make sure there is a signal in one of the display slots and at least three free displays. Select a signal.
- In Signal>Spectrogram Setup select "Black and White" and "Five-tone shading", turn "Separate Window" off. Make other selections as desired. Exit with "Spectrogram/Cepstrogram" and let Signalyze draw a spectrogram.
- Print the signal display and the spectrogram with the menu command File>Print.

- You can stop the printing with COMMAND-period.

High Resolution and Color

When the Macintosh II was released a few years ago, Apple came out with a sophisticated color image standard. This standard is incorporated in all Macintosh computers capable of color or grayscale. Here are some of its highlights.

- It theoretically permits grays or colors anywhere from simple black & white to 4.2 billion distinctions. Simple black & white is handled by single bits per pixel, and the 4.2 billion are handled by 32 bits per pixel. Popular offerings are 4 bits (16 grays or colors), 8 bits (256 grays or colors), and 24 bits (the current high end with 8.4 million grays or colors visible at any one time). Signalyze can use B/W, 16 colors/grays or 256 colors/grays.
- The user can select the exact black and white or color/grayscale shading through the "Monitors" Control Panel accessory in the Apple Menu.
- The more shadings are desired, the more memory is required. Conversely, the fewer shadings, the faster the display. If you use standard black & white, you use only one bit per pixel, and the display is the fastest. If you use 256 levels of gray or color, you use 8 bits (or a byte) for each visible pixel, and the display is a fair bit slower.
- The "old" Macintosh standard also had color, though few programs used it and the standard Mac can't show it. Signalyze can use old-standard color in the Signal Window and new-standard color in high-resolution spectrogram windows.
- The newer standard is fully compatible with and incorporates the previous Macintosh standard—at the cost of a few programming handstands.
- Signalyze uses color as follows:

  — On black-and-white Macintoshes, the *Signal Window* and the *Text Window* are *old-standard* windows, permitting only standard definition images with optional old-standard color.

  — On new-standard Macintoshes, the Signal Window, the *large spectrogram* and *plot windows* can optionally be *new-standard* windows.
- Windows are saved in their own standard, including color and grayscale images.

Printing a High-Resolution Spectrogram on a Separate Window

- To print without reduction from a large screen, you may have to reduce the size of the Signal Window first. Change the startup values with Edit>Startup Options to "Manual Window Sizing", six

displays per card, and set "Signal Display Width" to 554 pixels (as in an 11-inch screen). Exit the program and restart.
- Set the Monitors Control Panels accessory to 16 or 256 grays. 256 grays gives somewhat better spectrograms.
- If you're working with a large screen, select *sideways* display in File>Page Setup.
- If necessary, bring the Signal Window to the front.
- Select a signal.
- In Signal>Spectrogram Setup select "Grayscale", turn "Separate Window" on. Make other selections as desired. Exit with "Spectrogram/Cepstrogram" and let Signalyze draw a spectrogram.
- Print the signal display and the spectrogram with the menu command File>Print. In the print dialog, select "Color/Grayscale". If your print dialog does not have a "Color/Grayscale" option, you do not have a suitable print driver.
- If you're on an Apple Laser Printer, chances are that you'll now have to wait an excessive amount of time (about 10-25 minutes) for the spectrogram to print out the high-resolution spectrogram. Some other PostScript printers (e.g., a PostScript-equipped HP printer) sometimes take less time.
- To circumvent such excessive wait times, save the spectrogram as an image and print it out with any one of a number of graphics processing programs (e.g., "NIH Image"). These programs have special drivers that accelerate high-resolution image printing.

Printing a Plot Window
- Follow the printer setup instructions given above. Select sideways display. Make sure there's a signal available.
- In Spectral>Plot Setup, select "Time Plot".
- In the signal, blacken a small feature, such as a vowel onset.
- Do "Spectral>Make Plot". A new, large window is created with an enlarged version of the selected signal feature.
- Make sure the plot is in front.
- Print the plot with the menu command File>Print.

Printing a Text Window
- Follow the printer setup instructions given above.
- In Page Setup, select normal display orientation.
- Make sure the text window is in front.
- Type some text, or read in some text with File>Open.
- Print the text with File>Print. The text is printed and a page break is inserted automatically every 66 lines.

Startup Options

Most aspects of the dialog box Edit>Startup Options are self-explanatory. You can have the Signal Window come up in a certain place with as many display slots (beyond a minimal 4) as you want, or you can have the program fit it into the screen automatically. You can set the maximum number of cards and signals. Smaller numbers of such objects cut down on overhead and leave more memory for signals.

However, what needs explaining is the difference between *signal* and *calculation memory*. Because signals take so much memory and require much processor-intensive access, Signalyze manages its signal memory separately from all other memory. On the other hand, calculation memory is handled by the Macintosh memory manager.

Initially, Signalyze reserves a minimum amount of calculation memory and requests as much signal memory from the Macintosh memory manager as it can get. This is usually what is needed to maximize the signal work space.

However, if you often use 256-color/grays *high-resolution spectrograms*, this approach may put you in a memory squeeze. Increase *calculation memory*, exit the dialog, quit from the program and start up again. If you check the About dialog, you'll see that signal memory has decreased, and that calculation memory has increased.

You'll seldom have to change the *signal memory* request. However, if you get *crashes* while running a processor-intensive routine (e.g., splines or spectrograms), try reducing signal memory. Your system may be very large and the *stack* (system memory) may be bumping into *calculation memory*. The only way to prevent that is to reduce signal memory. For more details on memory handling, please refer to the Technical Information section.

Unsolved Memory Mysteries: The Macintosh memory manager does a great deal of work behind the user's (and often the programmer's) back. Resources are loaded in and purged, memory is compacted and rearranged, etc., etc. Consequently you may see more calculation memory in the About dialog at the start of the program than after awhile. So if you're getting strange memory squeezes, increase calculation memory, quit, restart, run something, and *then* check memory in the About dialog.

Working in Different Languages

As you worked through this Guided Tour, you may have noticed that you can easily switch Signalyze into French and German, and back to English through the Edit>Language dialog. The program's operation is exactly the same in the three languages, only the menus, dialogs and error messages are changed. The reference section lists the various commands in all three languages.

The two extra languages do not "eat up" any internal memory. Only one language resides (to some degree) in memory at any one time[1].

Those who are familiar with editing resources can translate Signalyze into yet other languages. Be aware, however, that translating all the dialogs and menus is a fair bit of work and will have to be redone as soon as the next Signalyze version comes out.

Batch Load

Let's look at a quick way of loading a whole series of signals at once.

- Return to the desktop.
- Create a new folder and put several Signalyze document files into it.
- Open the new folder and choose the Edit>Select All menu command (or do COMMAND-A). All file icons are selected and blackened.
- Choose File>Open (or do COMMAND-O). Signalyze starts up or comes to the front and loads all of the selected files in alphabetical order.
- Your files are displayed with the default values you specified the last time you used the program. That's because Signalyze "remembers" the default plot, spectrogram, file and most other settings from one work session to the next.

Where to Go From Here

Now that you've explored the most commonly-used Signalyze functions, you may wish to discover some of the remaining functions of the program. Please work with the Contents and Index pages to find the information you need.

---

[1] In MacSpeak: all language-specific dialog and string resources are marked purgeable.

Installation Suggestions

Recommended Hardware[1]

Computer
You need a minimum of 4 Mb of memory to run the current version of Signalyze. 8 Mb or more is desirable. For use with very large files, Signalyze can be run in virtual memory on computers that support the virtual memory mode. However, virtual memory mode is not compatible with some input and output devices.

The Macintosh Plus or Macintosh SE is the smallest computer on which Signalyze runs (operations are slow). Macintosh SE owners may want to buy an accelerator board to secure speed improvements like: 12 seconds instead of 50 seconds for a 427-pixel spectrogram; 1 instead of 4 seconds for a pitch extraction. Macintosh II (or SE-30, Portable, PowerBook, IIx, IIcx, IIci, IIfx, Quadra) performance is similar or better. Because of extensive use of integer math throughout the program, further speed increases on the more advanced machines (IIx, cx, vx, ci and vi) are not excessive, except in the case of splines and other floating-point transformations. The one exception so far is the Quadra. With all caches enabled, Signalyze operation is *very fast* on a Quadra (also, Signalyze is totally compatible with the Quadra). All Power Manager operations function correctly on the Macintosh Portable and on the various PowerBooks.

Signalyze also runs well on the new generation of RISC-based machines (Power Macintosh and descendants). These are very attractive machines because of their price-power relationship and because they have a built-in 16-bit sound capability. Also, Signalyze only exploits part of their built-in power at the present time. Some interesting speed gains will become available on such machines as parts of Signalyze are translated into RISC-native code.

To our knowledge, there is no *incompatibility with any of the named machines*. If you ever encounter an incompatibility with any of these machines, please contact InfoSignal Inc. We will do our best to clear up the problem.

---

[1] All indications valid when last verified. No responsibility taken for changes in hardware.

Disk and Backup Media

A large hard disk is recommended for serious signal analysis work (80 Megabytes or more, the more the better). Back-up media are so varied in price, availability and suitability that no firm recommendation can be made. Entry-level user can be assured that over years of lab use, floppy disks have turned out to be a fairly time-efficient and inexpensive long term storage medium. Removable hard disks are an attractive medium-price alternative. We have been less enchanted with tape media because of the excessive access time required on most tape operating systems.

Signal Acquisition

Serious speech research is possible with 8-bit signals acquired with the Macintosh built-in microphone or inexpensive MacRecorder or similar devices.

Improved spectrograms, much improved sound reproduction, and amplitude measures are possible with 12- or 16-bit signals acquired with Apple 16-bit, Digidesign, MacADIOS equipment, or with A/D-D/A devices produced by a number of other manufacturers.

Signalyze is set up to read sound from *any* System-7 compatible input driver. Please contact InfoSignal Inc. if your version of Signalyze cannot record sound via such an input driver. We'll work with you to insure compatibility.

At this writing, the most attractive, most reliable and most cost-effective 16-bit environment is the Power Macintosh. For long-term compatibility, "buying Apple" is still the best policy.

Signalyze Installation

(If necessary, please also consult the file "Installation Instructions" that came with your distribution disk. This file contains the latest updates that are appropriate to the version of the program you've received.)

Once you have established that you are willing to be bound by the License Agreement…

- Make sure you have a Macintosh with at least 4 Megabytes (4000 k) of internal memory (RAM). Signalyze 2.0 cannot run in less RAM. Any 4 Mb+ Macintosh, starting with the Macintosh Plus series and operating under System 4.2 or higher, can run Signalyze.
- Copy all files from the Signalyze source diskette onto a hard disk or another diskette. Never work directly on the distribution diskette.
- Store your distribution diskette in a safe place.

- Double-click on all compacted files to decompress their contents. Place the Signalyze program into the Signalyze Folder.
- If you're working in MultiFinder or under System 7.0, select (blacken) the Signalyze application icon and use the Get Info (COMMAND-I) menu command to set the memory request to the largest amount of RAM you can typically spare for Signalyze. A good amount is 4000k (4 Mb). The program can currently run (tightly) with a minimum of 1900 k of RAM (1.9 Mb).
- Open Signalyze by double clicking on the Signalyze icon or by performing the File>Open routine.
- If the Open operation fails...
  — If you're working under System 7.0 or MultiFinder, and Signalyze fails (it shouldn't), switch to "Finder" under system 6.0x. The program is large and has a massive appetite for solid blocks of memory. Your present standard shared-processing settings may not be adequate for Signalyze.
  — If you're working in Finder and get a crash when opening, your System file may be too large or you may be working with an incompatible System. Signalyze was tested and runs with a variety of systems on a large number of Macintoshes. (The Macintosh 128 and 512 are not supported.) In this case, you may wish to experiment with reducing the size of your System file by eliminating fonts and less-needed desk accessories. Or you may want to try Signalyze with an older or newer System file.
  — If you continue to get a system error, your diskette may have been X-rayed by the post office, or you may have incompatible hardware (e.g., a highly unusual accelerator board). In case of X-ray damage, InfoSignal will exchange your original distribution diskette for another one. If your failure is due to other reasons, InfoSignal will do its best to work with you to fix the problem (see License Agreement).
- When the program opens, turn to the section about Edit>Startup Options in the User's Reference Section of the manual, open the Edit>Startup Options dialog in the program, and choose your startup configuration.
- File>Quit Signalyze and start over again. Now your startup options will take effect.
- *Do yourself a favor and find some time to take the Guided Tour. Without it, you will not be aware of most of the program's options and you'll be left with many questions concerning the exact operation of the program's many buttons, click and menu options.*
- Finally, take some time to familiarize yourself with the remaining sections of this manual.

What Do You If You Don't Have Enough Memory?

The answer is—for most people—**use virtual memory (or RAM Doubler)**. Signalyze, like all signal analysis programs, needs lots of memory. Memory shortages are chronic. So if you have a computer capable of virtual memory (find out in the "Memory" Control Panel), do the following:

- Request a massive chunk of virtual memory and reboot.
- Alternatively, install "RAM Doubler" (a dynamo utility program from Connectix).
- Increase the maximum memory request of the Signalyze application to 8, 10 or 12 Mb of memory. (At the Desktop level, click on the Signalyze icon and do File>Get Info, then adjust maximum memory).
- Now you can compare large numbers of spectrograms and look at enormous signal files—all without that hideous "insufficient memory" message!

User Support

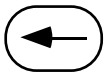
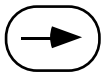
- *If you have a question about Signalyze, study the manual.* Consult the index. Look up the reference section. Nearly every aspect of the program is explained in this manual. In six years of doing support for previous versions of this program, very few answers to questions about the program were not available somewhere in a logical place in the manual.
- If after reasonable search something still isn't clear, please check out the "support" sections of www.infosignal.com.
- You can always get in touch with InfoSignal via InfoSignal@csi.com, but because of time constraints, no answer can be guaranteed.
- Sorry, there is *no telephone support.*

User's Reference Section

Index of this section:

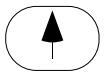
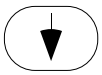
Signalyze keyboard and clicking actions
Buttons: Full descriptions
Scroll bars
Menu commands

Signalyze Keyboard and Clicking Actions

 **and**  **Left and Right Arrow Keys**

With the Signal Window in front and signal editing selected *:*
Moves the line cursor of a selected signal to the left and to the right one pixel at a time. Capable of repeat function when held down.

With the Text Window in front, or with the Signal Window in front and text editing selected / *:*
Traditional text cursor movement in the Text Window or in the Edit Window (upper left corner of the Signal Window).

 **and**  **Up and Down Arrow Keys**

With the Signal Window in front and signal editing selected *:*
Switch the selected signal to the signal lying directly above or below the presently selected signal.

With the Text Window in front, or with the Signal Window in front and text editing selected / *:*
Traditional text cursor movement in the Text Window or in the Edit Window (upper left corner of the Signal Window).

With the Text Window in front:
OPTION-UP/DOWN ARROW: Scroll whole page.
COMMAND-UP/DOWN ARROW: Smooth-scroll whole page.

(Set scrolling speed through Edit>Preferences).

The RETURN Key

With dialogs in front:
Acts as an equivalent of the framed button (the strongly-outlined button).

With Signal Window in front:
Sends textual material from the Edit Window to the Text Window.

Menu Commands ⌘M

Signalyze supports the Macintosh-standard COMMAND key equivalents of menu commands (given as ⌘-letter combinations in the menus). To execute such a command, hold down the COMMAND key (⌘) at the same time as you press the letter key (upper or lower case).

Key-click Combinations

Most common signal manipulation, signal scoring and editing actions can be invoked very rapidly with Signalyze-specified key-click combinations (i.e., clicks executed while holding down the buttons SHIFT, OPTION, COMMAND, or any combination thereof).

Where to Find Descriptions of Key-click Combinations

Descriptions of the principles of their assignment and of clicks executed in signal tracks are given here. Descriptions for key-click combinations with specific buttons are given under the button descriptions in the next section. A summary list of key-click combinations is found in Signalyze Help. Print it out for easy reference.

Key-click Combinations You Absolutely, Positively Do Not Want To Miss. Signalyze has a lot of key-click combinations—so many that you may not know where to start. Don't give up. Check out the "Hints" screen in "About Signalyze" and learn the few key click combinations shown at the bottom left. You won't be sorry you did.

Principles of Key-click Assignment

SHIFT — Gives popup menus, edit popouts, and dialog boxes.
OPTION/COMMAND — Gives binary options, e.g., narrow vs. wide band spectrograms.
COMBINATIONS OF KEYS — Shows values in Edit Window.

Key Presses at Startup

- Hold OPTION-COMMAND: Put Signalyze into read-only mode.

Clicks in Signal Display

- Simple click: Line cursor placement.
- SHIFT-click: Extend signal selection.
- SHIFT-COMMAND-click: Add time indication to Edit Window.
- SHIFT-OPTION-COMMAND-click: Show bottom number of Signal Control Panel in Edit Window.
- SHIFT-OPTION-click: Erase Edit Window.
- Double-click: Transfer Edit Window to Text Window, add Return.
- OPTION-click: Zoom left to cursor.
- COMMAND-click: Zoom right to cursor.
- OPTION-COMMAND-click: Displace left edge of display to cursor position.

Clicks in Mini-Display

- OPTION-click: Zoom left to cursor.
- COMMAND-click: Zoom right to cursor.
- OPTION-COMMAND-click: Displace left edge of display to cursor position.

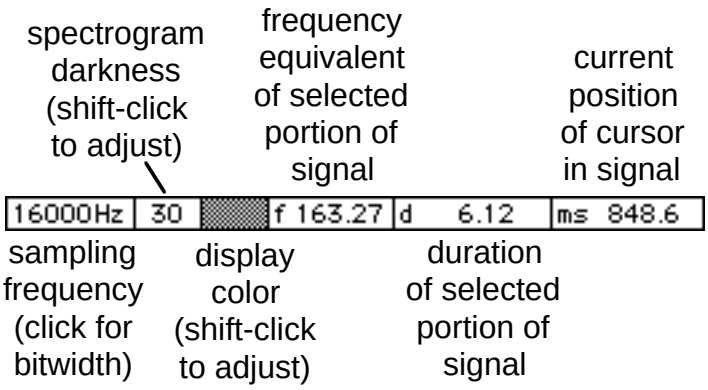
Clicks in Spectrogram/Cepstrogram

- Simple click: Line cursor placement.
- SHIFT-click: Extend signal selection.
- SHIFT-COMMAND-click (Spectrogram): Show time and frequency in Edit Window.
- SHIFT-COMMAND-click (Cepstrogram): Show time, f0 cycle length, and frequency in Edit Window.
- SHIFT-OPTION-click: Erase Edit Window.
- Double-click: Transfer Edit Window to Text Window, add Return.

Clicks in Spectrum/Cepstrum

- Simple click: Line cursor placement
- SHIFT-click (Spectrum or Cepstrum): Show frequency and time where spectrum taken in Edit Window
- SHIFT-OPTION-COMMAND-click (Spectrum): Show dB in Edit Window
- SHIFT-OPTION-COMMAND-click (Cepstrum): Show Fo cycle duration in Edit Window
- SHIFT-OPTION-click: Erase Edit Window
- Double-click: Transfer Edit Window to Text Window, add Return

INFORMATION BAR



Clicks on Information Bar Indicators

— Frequency/Bit width Indicator (leftmost)

- No key: Toggle between current signal's sampling frequency and bit width.
- SHIFT: Edit sampling frequency.

— Spectrogram Darkness Indicator (2nd)

- SHIFT: Set spectrogram darkness.

— Color Indicator (3rd from left) (only operational on Macs with color/grayscale). The palette supports "old-standard colors" (see Tutorial 8).

- SHIFT: Set color of selected plot.

— Frequency Indicator ("f", 4th from left)

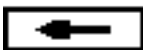
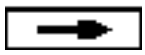
- No key: Show frequency equivalent of duration of signal selection in Edit Window. Only operational if duration > 0. (Frequency equivalent = 1000 / duration in ms).

— Duration Indicator ("d", 5th from left).

- No key: Show duration of signal selection in Edit Window

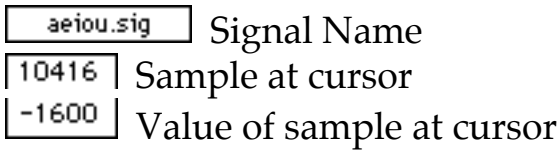
— Time Indicator ("s/ms/μ", 6th from left).

- No key: Show time at the cursor in Edit Window.

— Card Indicator (between  and  buttons)

- No key: Jump to card x.

SIGNAL CONTROL PANEL INDICATORS



Signal Name
Sample at cursor
Value of sample at cursor

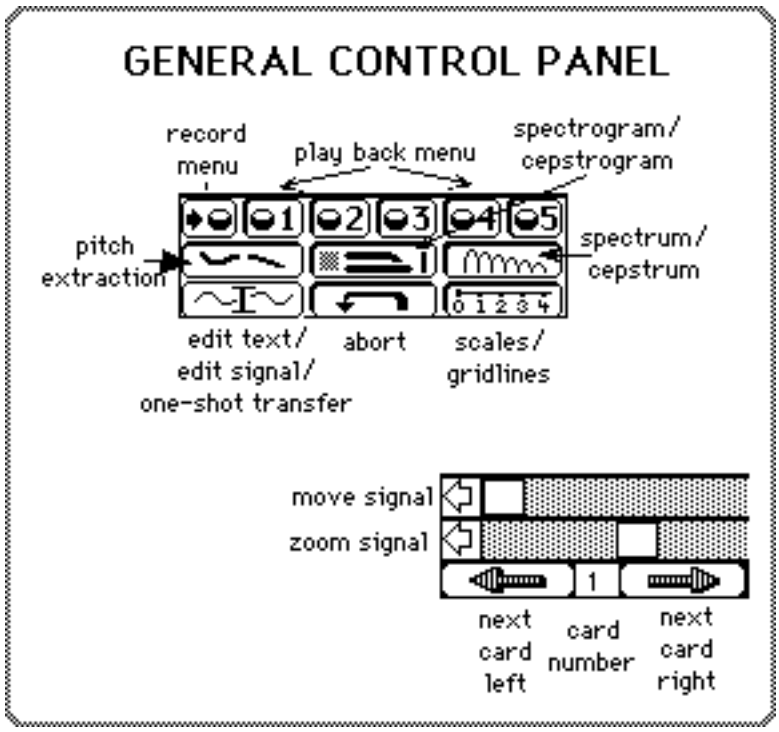
Clicks on Signal Control Panel Indicators

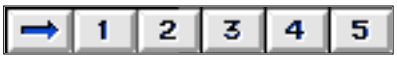
— Signal Name (top row)
- No key: Show name in Edit Window
- SHIFT: Edit signal name

— Top Number (second row)
- No key: Show top number in Edit Window

— Bottom Number (third row)
- No key: Show bottom number in Edit Window
- SHIFT: Edit bottom number (regular signals only)

Buttons: The General Control Panel





Record Menu Command

The button initiates the recording of a menu command. After clicking on this button, follow the instructions in the Edit Window. Click on one of the Play Back Menu Command buttons ( ), and select a menu command. After recording the menu command, the button will "play it back" when clicked. The menu command is remembered until it is replaced with another. For technical reasons, you cannot store a command from the -menu, nor the Macro>Play Sequential Macro command.

Menu command buttons can be stored in the sequential macro facility (Macro>Sequential Macro Setup). Rule: The sequential macro can call a menu command button but not vice versa. This is done to prevent infinite loops.

Letter equivalents: To invoke these menu commands very rapidly, you can store letter a equivalent for each of the five buttons. The storing routine is found in Edit>Preferences. *In this way you can make up to five additional Signalyze menu commands keyboard-callable.*

1 to 5 Play Back Menu Commands

After a menu command is recorded with the Record Menu Command button, these buttons play back the command (one command per button). To find out which menu command is stored in a Programmable Function button, click on the button while holding down either the SHIFT or the OPTION key. The name of the menu command appears in the Edit Window (in the language in which it was originally stored, e.g., in English, if it was stored in English, even though you may be in the German user interface).

Three Ways To Stop Execution:
1. Press COMMAND-period.
2. Press COMMAND, then click.
3. Click on the ABORT button ( ).

Pitch Extraction Button

Performs a pitch extraction according to specifications in the Pitch Extraction Setup (equivalent of Spectral>Extract Pitch).

- SHIFT**:** Calls Spectral>Pitch Extraction Setup.

Spectrogram/Cepstrogram Button

Performs a spectrogram or cepstrogram according to specifications in Spectral>Spectral Analysis Setup (equivalent of Spectral> Spectrogram/ Cepstrogram). Wide-band spectrograms/cepstrograms are based on the results of a 256-point FFT and narrow-band spectrograms/ cepstrograms use 512-point FFTs.

- SHIFT: Calls Spectral Analysis Setup.
- OPTION: Specifies wide band spectrogram.
- COMMAND: Specifies narrow band spectrogram.

Spectrum/Cepstrum Button

At the place indicated by the cursor, this routine performs a 1024-point spectrum or cepstrum ("slice plot", or "section") according to specifications in Spectral > Spectral Analysis Setup.

- SHIFT: Spectral Analysis Setup

, A B and Edit Buttons

If the button is active, the Menu Commands "cut", "copy", and "clear" apply to the currently selected signal (marked 1), and the

Edit Menu command "paste" applies to the display track indicated by the destination button ( ).

If the or button is active, the Menu Commands "cut", "copy", "paste" and "clear" apply to text in the Edit Window.

Switching Between ***and*** ***Buttons***

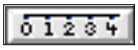
Clicking on these buttons switches between signal text editing modes. Switching also occurs upon clicking in the appropriate edit area (in the Edit Window or in the signal tracks).

More on the ***Button: "One-shot Transfer" Mode***

The one-shot transfer mode can only be entered by toggling through the Edit Buttons. If selected, "Edit>Copy" causes the entire contents of the Edit Window to transferred to the Clipboard, from where it can be inserted into another program. This way, a whole series of TAB-delimited numbers can be transferred into another application in "one shot". This edit mode is not sensitive to clicks, and stays active until another edit mode is selected.

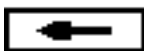
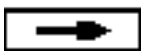
Abort Button

Used to abort a multi-step command or nearly any ongoing process.

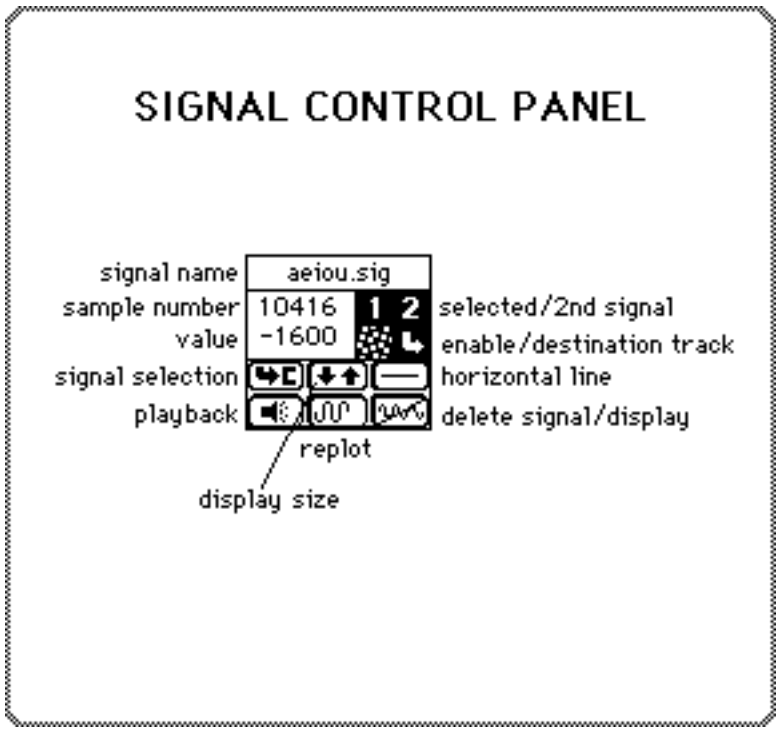
 **Scales Button**

This button toggles scales and gridlines on and off according to specifications in Signal>Scale Setup.

- SHIFT: Calls scales dialog box
- OPTION: Toggle scales for y-axis
- COMMAND: Toggle scales for x-axis
- SHIFT-OPTION: Toggle gridlines for y-axis
- SHIFT-COMMAND: Toggle gridlines for x-axis

 **and**  **Card Switch Buttons**

Used to switch to next lower-numbered or next higher-numbered signal card. For jumping to remote signal cards, try clicking on the number between the Card Switch buttons.



Buttons: The Signal Control Panel

1 **or** 1 **Selected Signal Button**Identifies which of the signals on the card is the *selected signal.* (Most operations only apply to the selected signal). A new signal can be selected by clicking on this button, by clicking anywhere in the signal tracks, or by activating the UP or DOWN keys on the keyboard.

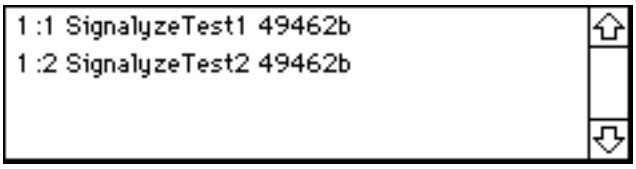
2 **or** 2 **Second Signal Button**
Identifies which of the signals on the card serves as *second signal* in a simple transformation involving a second signal. See the Transform>Simple Transformation menu command.

Enable Button

With Labels		Without Labels	
	active		active
	passive		passive

When the Enable Button appears active, zooms and signal displacements apply to the signal displayed to the right. When it appears passive, the signal remains stationary even when other signals are modified. *This way you can compare two different time segments, despite Signalyze's inherently synchronous display characteristics.*

or **Destination Track Button**Identifies the *destination track* for signals loaded from a file or generated through any of the program's processes. When this button is activated next to an unused signal track, new signals are loaded into the indicated track. When it is activated next to an occupied signal track, new signals are loaded into the next available track (set exact location through "Location in Signal Display" in Signal>Display Setup).



Signal Selection Button
Produces a small dialog box superimposed on the Edit Window listing all signals. Double-click on one of the signals, and the selected signal is inserted in the signal track at right. Click anywhere outside the dialog box to abort. The first number in front of the name indicates the card, and the second number indicates the signal track. Signal length is given in bytes after the name.

- SHIFT: Load signal via Open File dialog
- OPTION: Load signal via Open Sound Resource dialog

Display Size Button
Toggles between normal-size (64-pixel) and double-size (128-pixel) track displays.

- SHIFT: Sets the vertical zoom. Use large zoom settings to inspect detailed aspects of the signal.

Horizontal Line Button
Activates the horizontal cursor(s) according to specifications in Signal>Scale Setup. A second click removes the cursor(s). Normal-size displays have a single horizontal line cursor, and double-size displays have two.

- OPTION: Horizontal cursor follows click.
- COMMAND: Horizontal cursor stays stationary.

Sound Playback Button
Initiates sound playback of the signal displayed to the right according to specifications in Signal>Audio Setup.

- SHIFT: Set playback volume level.
- OPTION: Play whole signal.

- COMMAND: Play selected segment.
- OPTION-COMMAND: Play signal except for selected segment.
- The routine adjusts automatically to any significant bit width from 8-16 bits. It can play signals recorded at any sampling rate upwards of 5 kHz. With some routines on the slower Macs, there is a small delay prior to initiating playback.

Signal Replot Button

Replots the display of the signal shown to the right. For this command to be effective, the Replot Button must be on ( ).

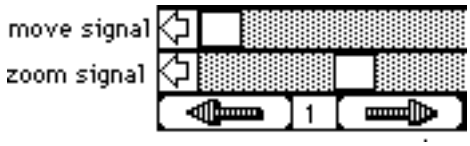
- OPTION: Replot as connected line plot.
- COMMAND: Replot as MinMax plot.
- SHIFT-COMMAND: Replot whole signal at display width.
- OPTION-COMMAND: Replot selected segment at display width.
- SHIFT OPTION: Toggle between normal/narrow plot (narrow plots are useful for showing pitch extractions).

Delete Signal/Display Button

Deletes the **signal**. Check out the various commands in the Signal menu to delete signals in other ways.

- **Displays** can be deleted without erasing the underlying signal by holding the COMMAND key while clicking on this button. Signals temporarily eliminated in this way can be reestablished with the Signal Selection Button ( ).

Scroll Bars



Move Signal

Causes the *left edge* of all enabled signal displays to move in the same direction as the scroll bar. Pressing in the arrow space causes a 1% edge movement, and pressing in the gray portion (the page left or page right portion) of the scroll bar causes a 3% edge movement.

On 9-inch Macintosh screens, it is suggested that the Left Edge Scroll Bar be used primarily for gross signal displacement, while option-command-clicks in the display space or the Mini Display can be used for fine left-edge adjustment. On wider screens, scroll bar adjustments can be used successfully to perform both gross and fine adjustments of the display's left edge.

Zoom Signal

- RIGHT: Zooms in (enlarges) all enabled signals.
- LEFT: Zooms out (decreases) all enabled signals.

The zoom is controlled by a power function; pressing in the arrow space causes a 1% change in the input to the function, and pressing in the gray portion (the page left or page right portion) of the scroll bar causes a 3% change in the input to the function.

On 9-inch Macintosh screens, it is suggested that the Zoom Scroll Bar be used primarily for gross signal displacement, while option-clicks and command-clicks in the main display or in the Mini Display can be used advantageously for precise signal positioning. For details, see Tutorial 1 of the Guided Tour.

Menus

Letters after the cloverleaf symbol (⌘) indicate **command keyboard equivalents.** E.g., **File>Open ⌘O** means that the **File>Open** can also be executed by holding down the command (cloverleaf) key while typing 'o'.

"Apple" Menus

>About Signalyze...

>Concernant Signalyze...

>Über Signalyze...

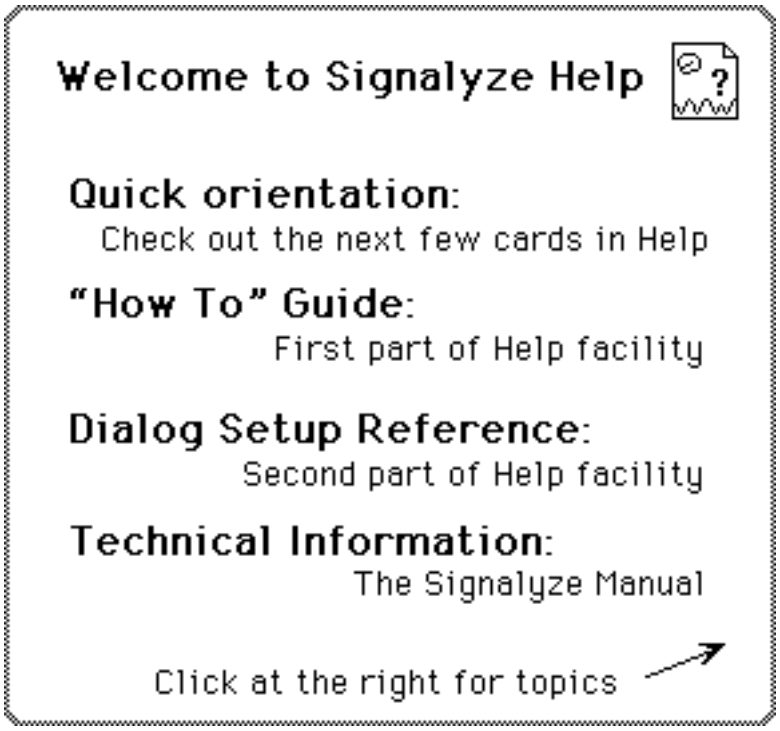
Displays information about the present version of Signalyze: the copyright notice, credits, the expiration date, plus current and attributed signal and calculation memory. Check Edit>Startup Options for notes on *requested* memory usage.

- The *"Hints "* button brings up a screen with the names of the various Signal Window buttons, plus a reminder of how to zoom.
- The *"Help"* button is equivalent to selecting >Help.
- For further information about memory usage, see "Memory Management" in the Technical Information section.

The "Hints" page can be printed out by clicking on this button.

>Help... ⌘-H
>Aide... ⌘-H
>Hilfe... ⌘-H

Transfers control to the Signalyze Help facility. For this to work right, the Signalyze program must be in the same folder as Signalyze Help.



The Signalyze Help facility contains three sections:

- *A few orientation cards:* A quick, graphic orientation of Signalyze objects.
- *A "How To" Guide:* Small recipe-type instructions how to perform most actions in Signalyze. Many of these instructions also appear in the Tutorials of this manual, amplified with suitable illustrations.
- *A Reference Section:* A full reference for each of the program's setup dialogs.

Any Signalyze Help page can be printed out by clicking on this button.

The Signalyze Menu Color Coding (color Macs only):	
Blue menu commands:	Input functions
Brown menu commands:	Housekeeping functions
Red menu commands:	Setup functions
Green menu commands:	Output functions

"File" Menus

File>New Signal ⌘N
Fichier>Nouveau Signal ⌘N
Ablage>Neues Signal ⌘N

With the Signal Window in front: Selects the next card with a free first slot. If the present card is free of signals, no action is taken.

File>New Text ⌘N
Fichier>Nouveau Texte ⌘N
Ablage>Neuer Text ⌘N

With the Text Window in front: The existing text is erased (after asking if it's Ok to do so) and an empty text window is presented. If the Text Window is empty, no action is taken.

File>Open Signal... ⌘O
Fichier>Ouvrir Signal... ⌘O
Ablage>Signal öffnen... ⌘O

With the Signal Window in front: Via the standard Macintosh file dialog, this command lets you select and load a file of the type specified through "Formats" popup or through the File>File Setup command.

- A *key-click equivalent* for this command is SHIFT-click on the Signal Selection Button ( ).
- For rules concerning where the file will be placed, see the information box "The Logic Behind Signal Placement", in Tutorial 6.

File>Open Text... ⌘O
Fichier>Ouvrir Texte... ⌘O
Ablage>Text öffnen... ⌘O

With the Text Window in front: Via the standard Macintosh file dialog, this command opens TEXT files. In the Text Window, it replaces any previously loaded text (after asking if it's Ok to do so). Only files less than 32k long can be loaded. For more information about the text editor, see "The Signalyze Text Editors" in the Technical Information section.

File>Save Signal ⌘S
Fichier>Sauver Signal ⌘S
Ablage>Signal Speichern ⌘S

Saves the currently selected signal. The command takes no action if the signal has not been changed since it was opened or last saved. If the signal has never been saved, this command is equivalent to the File>Save As... command.

- The file is saved using the *last-specified file format*. To change file formats, use File>Save As.
- For formats with *fixed sampling rates*, the closest sampling rate to that of the selected signal is chosen, subject to the "Output Tolerance" setting specified in File>File Setup. See File>File Setup for more information about Output Tolerance.
- *8-bit files* (SoundEdit and SoundWave files) require space in calculation memory for the 16-to-8 bit recoding operation prior to read-out to the file. If you get memory limitation messages, increase *calculation memory* through Edit>Startup Options.
- *Sound resources* cannot be saved with this command. Use the File>Save Sound Resource command.

File>Save Text ⌘T
Fichier>Sauver Texte ⌘T
Ablage>Text Speichern ⌘T

This command saves the current text to the currently active TEXT file, if it has been changed since it was opened or the last time it was saved. If the text has never been saved, this command is equivalent to the File>Save As... command. This command can be activated no matter which window is in front, so you can save your results regularly.

File>Save Signal As... ⌘W
Fichier>Sauver signal sous... ⌘W
Ablage>Signal speichern als... ⌘W

With the Signal Window in front: This command saves the selected signal under the specified name according to the file format specified through the Format popup or the File>File Setup dialog. Files saved in this manner can be read by the appropriate program (e.g., a file saved in AudioMedia format can be read by the AudioMedia program).

- You can save *"on top of"* an existing file with a new format. The old file (same name, old format) is deleted before the new file (same name, different format) is saved to disk.
- You can get into the File>File Setup dialog from the standard file save dialog. Click on the *"File Setup..." button* .
- For formats with *fixed sampling rates*, the closest sampling rate to that of the selected signal is chosen and the signal is up- or down-sampled appropriately. Degree of permitted up-/down-sampling is subject to the "Output Tolerance" setting specified in File>File Setup. See File>File Setup for more information about Output Tolerance.
- *8-bit files* (SoundEdit and SoundWave files) require space in calculation memory for the 16-to-8 bit recoding operation prior to read-out to the file. If you get memory limitation messages, increase *calculation memory* through Edit>Startup Options.
- *Sound resources* cannot be saved with this command. Use the File>Save Sound Resource command.

File>Save Text As... ⌘W
Fichier>Sauver texte sous... ⌘W
Ablage>Text speichern als... ⌘W

With the Text Window in front: This command saves the text in the Text Window as a TEXT file under the specified name. Files saved in this manner can be read by any Macintosh program capable of reading a TEXT file, e.g., MacWrite, Word, Edit, etc.

- Files are saved with the *creator signature* specified in the Edit>Preferences menu.

File>Open Sound Resource... ⌘R
Fichier>Ouvrir ressource sonore... ⌘R
Ablage>Tonressource öffnen... ⌘R

This command opens a dialog box listing all folders and files. If you select a file (e.g., a HyperCard stack) in which a sound resource has been recorded, a selection dialog is superimposed on the Edit Window that lists all stored sound resources and their length in bytes. You can cancel the operation by clicking anywhere outside the dialog. If you double-click on a listed sound resource, the signal is loaded according to the sound resource format specified in File>File Setup.

- The routine reads 'snd ' resources of *standard formats 1 and 2* (mono, no compression, a single block of samples), plus a "generic" format.
- A *key-click equivalent* for this command is OPTION-click on the Signal Selection Button ( ).
- If the *sampling rate* is erroneous, try a different sound resource format via the File>File Setup dialog box. If that still fails, adjust the sampling rate via Signal>Signal Information or through SHIFT-click on the sampling frequency indicator.
- Since sound resources must be read into Calculation Memory in a single operation before being stored in Signal Memory in 16-bit format, reserve enough *calculation memory* for your sound resource. There's no way around this buffer operation, given the 8-bit nature of sound resources and Signalyze's 16-bit internal structure (see Technical Information section).
- For the rules governing the slot where the file will be displayed, see the box "The Logic Behind Signal Placement" in Tutorial 6.
- For technical reasons, the *System file* cannot be opened with this command.

File>Save As Sound Resource
Fichier>Sauver comme ressource sonore
Ablage>Als Tonressource speichern

Saves the selected signal as a standard Format 2 'snd '-resource in a specified file (this is the format compatible with HyperCard). Once in the "Save as a Sound Resource" dialog, you can create a System 7 sound resource file to store the signal. Alternatively, you can select any other destination file (such as a HyperCard stack) for storing the sound resource.

You can store several sounds in a "System 7 sound". The first sound you save in the file is the one that is played back when you double-click

the file under "Finder". Further sounds stored in the same file can be read by Signalyze and other programs capable of reading sound resources.

- The standard Apple sound resource format does not allow for *sampling frequencies above 32767 Hz*. High frequency signals thus should be de-sampled to a lower frequency (Transform>Complex Transformation) before they are stored as a sound resource.
- Since sound resources are read out to file in a single operation, they necessarily require space in calculation memory for the 16-to-8 bit recoding operation prior to read-out to the file. If you get memory limitation messages, increase *calculation memory* through Edit>Startup Options. There's no way around this buffer operation, given the 8-bit nature of sound resources and Signalyze's 16-bit internal structure (see Technical Information section).

File>Open Image

Fichier>Ouvrir image
Ablage>Bild öffnen

Opens PICT images saved by Signalyze or other programs in their native, color/gray characteristics.

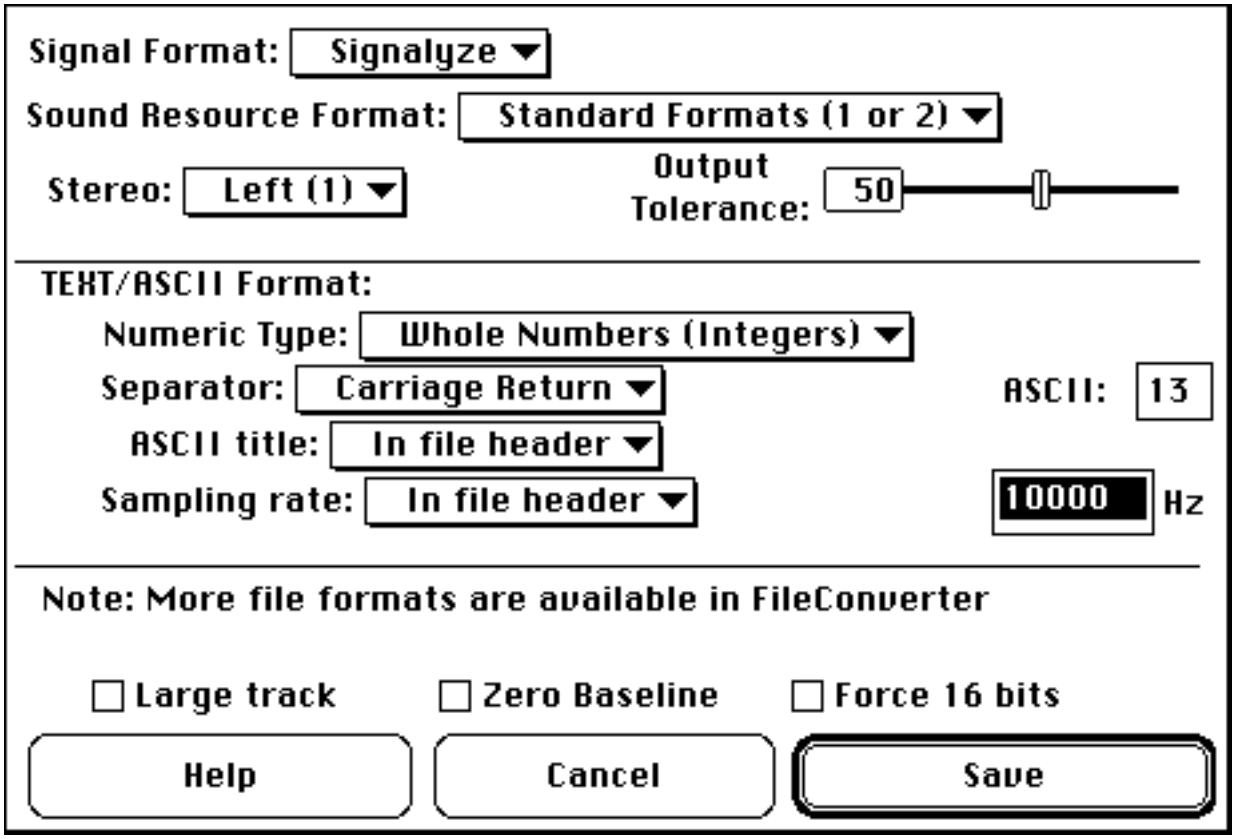
File>Save Image

Fichier>Sauver image
Ablage>Bild speichern

Saves current window as a PICT image in its native, color/gray characteristics.

File>File Setup...

Fichier>Options de fichier...
Ablage>Optionen Dateitypen...



This dialog box lets you select the format used by the read and write operations for signal files.

- ***Signal Format:*** Standard signal read and write formats built into Signalyze. These are the formats that are operational during the File>Open, File>Save and File>Save As menu commands. Translators for more signal formats are found in the FileConverter program supplied with Signalyze. For technical details about these formats, see below and the Technical Information section of this Manual.
- ***Sound Resource Format:*** Operational during the File>Open Sound Resource and File>Save Sound Resource menu commands. See below for more details.
- ***Stereo:*** Specifies input channel(s) for stereo files (currently: Digidesign and SoundEdit formats).
- ***Output Tolerance:*** Used in save operations to protect against excessive up- or down-sampling. It works as follows: Many sound processing programs only support certain sampling rates. For example, MacSpeech Lab only supports the sampling rates 83,332 Hz, 41,666 Hz, 20,833 Hz, 10,416 Hz, and 5,208 Hz. Signals having different sampling

rates have to be "forced" into the nearest set sampling rate when saved. E.g., if a 22 kHz signal is to be saved as a MacSpeech Lab file, it has to be converted into a 20 kHz file involving a 10% change in sampling frequency. This "forcing" involves up- or down-sampling, i.e., the regular duplication or skipping of sample values. The tolerance level serves as a safeguard that only signals with reasonably similar sampling rates are converted. If a conversion beyond the specified tolerance level is attempted, an alert box is produced and the conversion is aborted.

- ***Text/ASCII Format:*** Operational during the File>Open, File>Save and File>Save As menu commands to provide import/export functions for signals in TEXT format.

— *Numeric Type:* Input/output can be either whole or decimal numbers from -32,767 to +32,767. However, the decimal portion of a decimal number is disregarded on input and set to zero on output. Also, numbers exceeding this range are set to -32,767 and +32,766 respectively (+32,767 = "missing value").

— *Separator:* Any ASCII character can serve as *single* separator. Multiple separators (e.g. multiple tabs or multiple blanks) are not supported. *Special rules for blanks (ASCII 32):* Usually, blanks are stripped from ASCII numbers on input. However, if the blank serves as a separator, it must appear as *single* character between ASCII numbers, otherwise the routine will be "confused".

— Since *commas* are not recognized by Think C, they cannot set off thousands. A number like "1,234.05" could be treated as two signal values, "1" and "234.05". Eliminate commas with a word processor before reading numerical values with Import Signal.

— *Title and sampling rate:*

If in header: Must appear as first and second item in file.

If not in header: Are either supplied by the user via a special dialog at the beginning of the read operation or are taken from the file name and the edit box in the dialog.

— Signalyze estimates at the beginning of the loading operation how much *memory* will be required for the signal. However, since Signalyze has no way of knowing *exactly* how long the signal will be in binary form, you may encounter a memory error in the middle of a fairly long upload. It is thus important to leave sufficient free signal memory.

- ***Zero Baseline:*** If set, the mean value of incoming files is determined, and the entire signal is offset so that the mean value becomes zero. Since most digitized input consists of positive values only, this usually means that the signal changes from positive-only (or "unsigned") status to positive-and-negative (or "signed") status.

Note: In contrast to some other signal analysis programs, Signalyze performs zero baseline offset automatically for FFT analysis. Leaving this

option off therefore poses no problem. Spectral analyses are slightly faster when performed on signals set to zero-baseline by this option.

- ***Force 16 bits:*** This lets you input a 16-bit file exactly as it was recorded and no normalization will be applied. Suppose for example that you recorded an AudioMedia file with the AudioMedia program. If the sound volume was low throughout the file, Signalyze would usually determine a bit range less than 16 bits (perhaps 15 or 14 bits). If you then save the file again in AudioMedia format, the sound level would appear to be boosted so as to fill the entire 16-bit range. But if you set the "Force 16 bits" option, the file will appear in Signalyze exactly as it appeared in the AudioMedia program, and vice versa. The cost you pay for forcing a file into 16 bits is that the sound may appear weak on playback and that spectrograms need much more darkening than usually.

More on Signal Formats

12 / 16-Bit Formats: These files use two bytes (16 bits) for each data point. Some files in this format use all 16 bits for data information, while others use only 12 bits for data and leave the remaining four bits unused.

- ***Signalyze Format*** from InfoSignal Inc. is similar to the MacSpeech Lab format (see below), except that it permits a larger range of sampling rates and bit widths (1 Hz to 32 megahertz, 8-16 bits). Sampling rates up to 32,767 Hz are specified with a precision of 1 Hz, while sampling rates above 32,767 Hz are specified with a precision of 1 kHz. Maximum sampling rate: 32.766 Megahertz, minimum sampling rate: 1 Hz.
- ***MacSpeech Lab Format*** uses a system-specific code for each of the sampling rates permitted by the MacSpeech Lab programs from GW Instruments Inc.
- ***Sound Designer II Format*** uses a resource-specified number for each of the sampling rates permitted by the AudioMedia and Sound Designer II programs from Digidesign Inc. Can be in stereo.

8-bit Formats: These files use one byte (8 bits) for each data point. An 8-bit data point can only distinguish 256 gradations, in contrast to the 65,535 possible with 16-bits. 16-bit files have more precision for frequency analysis and suppress hiss better, while 8-bit files take up only half as much space.

- ***SoundEdit / MacRecorder Format:*** Uses a system-specific code for each of the sampling rates permitted by the SoundEdit program produced by MacroMind, Inc. Can be in stereo.
- ***SoundWave / Impulse Format***: Uses a system-specific code for each of the sampling rates permitted by the SoundWave program produced by the now defunct Impulse Inc.

TEXT/ASCII Format: These files are generally used for exchanging data with other programs and other computers (also called *Import/Export* format in some programs). Read operations in this format ignore all characters except: separator character, minus sign, decimal *period* (no decimal comma!), and ASCII numerals.

- Acceptable numerals are any integer from -32766 to +32766. Larger or smaller numbers are set to these limits.

Sound resources: These are 8-bit sound files stored *inside* another file. This makes it possible to keep many different sounds in a single file. If such sounds are stored in HyperCard stacks, they can be reproduced over the internal loudspeaker with the simple HyperCard "play" command. Signalyze can read sound resources stored in any file, including HyperCard stacks. To read a sound resource, you must first choose the file where the resource is stored, then the resource itself.

- ***Standard formats:*** Most sound resources use the two standard formats, "Format 1" and "Format 2". They conform to 1988 Apple standards for sampled-sound resources (see section on "File Formats" in Technical Information). If the playback doesn't seem right, adjust the sampling rate via the Signal>Signal Information dialog box or via the popup you get by SHIFT-clicking on the sampling rate indicator.
- The *Open Sound Resource* routine can read the standard format 1 and format 2 'snd ' resources, plus a "generic" type "B" format.
- The *Save Sound Resource* routine generates only format 2 resources, which is the standard format originally recommended by Apple for sampled sound and is the format used by HyperCard.
- ***"Generic" Format:*** Try this less common format if you don't seem to get the right sampling rate with the Standard Format option, or if a digitally sampled resource cannot be loaded with the standard format. This format assumes a default sampling rate of 22 kHz. If the playback doesn't seem right, adjust the sampling rate via the Signal>Signal Information dialog box or via the popup you get by SHIFT-clicking on the sampling rate indicator.
- Since sound resources are read and written from and to files in a single operation, they require memory in two places, first in calculation memory for the resource, and secondly in signal memory for a 16-bit version of the 8-bit sound resource. If your input file is very large (e.g., greater than 50k), you may have problems fitting it into calculation memory. If memory limits are encountered during the read and conversion process, increase the amount of *calculation memory* via Edit>Startup Options.
- If the resource does not consist of digitally sampled data, you may get a meaningless display or even a system crash.

File>Page Setup...
Fichier>Options d'impression...
Ablage>Druckereinstellung...

This command brings up the standard dialog box to set printing options for subsequent printing operations. The exact dialog box depends on the printer that has been selected through the **Chooser** desk accessory. Any printer that conforms to Apple standards for bit-mapped printing (such as the ImageWriter, the StyleWriter or the LaserWriter) can be used.

- After switching printers through the *Chooser* accessory, select this menu command and press "Ok" to inform Signalyze of the change (e.g., after changing from the ImageWriter to the LaserWriter). If you don't go through this little routine, Signalyze may attempt to route printing to the printer that was in operation when the program was started.
- Users of large screens may wish to use the *sideways display* and/or *reduction feature* to be able to print the contents of the entire signal display area.
- For further hints on printing, especially *fast* printing of spectrograms, see Tutorial 8.

File>Print... ⌘P
Fichier>Imprimer... ⌘P
Ablage>Drucken... ⌘P

- *If the **Signal Window** is in front:* Causes printing of the display area of the Signal Window. If the visible signal space exceeds a space about 8-by-10-inch square (20.3 cm by 54 cm), the print-out is clipped. Be sure you have a *printer driver* on your disk before initiating printing.

> **Space Wars:** The Macintosh printing manager needs a fair bit of Calculation Memory for this operation. On a Macintosh with a 9-inch screen window, it needs about 15 k, a number that can increase substantially with large screens and color/grayscales. If you get the "Insufficient Memory" message or even crashes during a print operation, increase the *Calculation Memory* allocation in Edit>Startup Options.

- *If the **Text Window** is in front:* Causes standard printing of the contents of the Text Window in the selected font and size. Page breaks are inserted according to the paper size setting selected in the Page Setup dialog.

File>Quit ⌘Q
Fichier>Quitter ⌘Q
Ablage>Beenden ⌘Q

Exits Signalyze and returns to the desktop.

"Edit" Menus

Edit>Cut ⌘X
Edition>Couper ⌘X
Bearbeiten>Ausschneiden ⌘X

With **(Signal Edit)**
"Cut signal" operation, performed on the blackened section of the currently selected signal. Signal segments are saved to a temporary buffer within Calculation Memory.

or **(Text edit in Signal Window or in Text Window)**
Standard cut text operation. "Cutting text" means taking material out and saving it to a buffer. Maximum number of characters to be cut: 10,000.

Edit>Copy ⌘C
Edition>Copier ⌘C
Bearbeiten>Kopieren ⌘C

With **(Signal Edit)**
"Copy signal" operation, performed on the blackened section of the currently selected signal. Signal segments are saved to a temporary buffer within Calculation Memory.

With **(Text edit in Signal Window or in Text Window)**
Standard copy text operation. "Copying text" means saving it to a buffer without taking it out. Maximum number of characters to be copied: 10,000. Copying text to other applications is supported under both Finder and MultiFinder.

With **("One-shot" transfer edit in Signal Window)**
Copies entire contents of Edit Window to a buffer. This permits to transfer a line of values to another program in "one shot".

Edit>Paste ⌘V
Edition>Coller ⌘V
Bearbeiten>Einsetzen ⌘V

With **(Signal Edit)**
"Paste signal" operation. A signal segment previously "cut" or "copied" is always inserted in the signal track with the **active destination button (** **)**:

- If the display slot is *empty*, the signal segment is inserted at the *correct time-aligned place.*
- If the display slot contains a *signal*, the signal segment is inserted at the *line cursor.*
- If the display slot contains a *blackened section of a signal,* it replaces the blackened section.
- Paste operations are optionally *smoothed at the transition.* See Edit>Signal Edit.
- Signal segments are copied from a temporary buffer within *Calculation Memory.*

With **or** **(Text Edit)**
Standard paste text operation, operational in both the Edit and the Text Window. Text previously saved to a buffer is inserted at the blinking insertion point or replaces blackened section of text.

- Maximum number of characters to be pasted: 10,000. If the text to be inserted brings the total text to over 32,000 characters, the operation is abandoned.
- Copying text from other applications is supported under both Finder and MultiFinder.

Edit>Clear ⌘B
Edition>Effacer ⌘B
Bearbeiten>Löschen ⌘B

With **(Signal Edit)**
"Cut signal" operation, performed on the blackened section of the currently selected signal, with the difference that the signal section is lost.

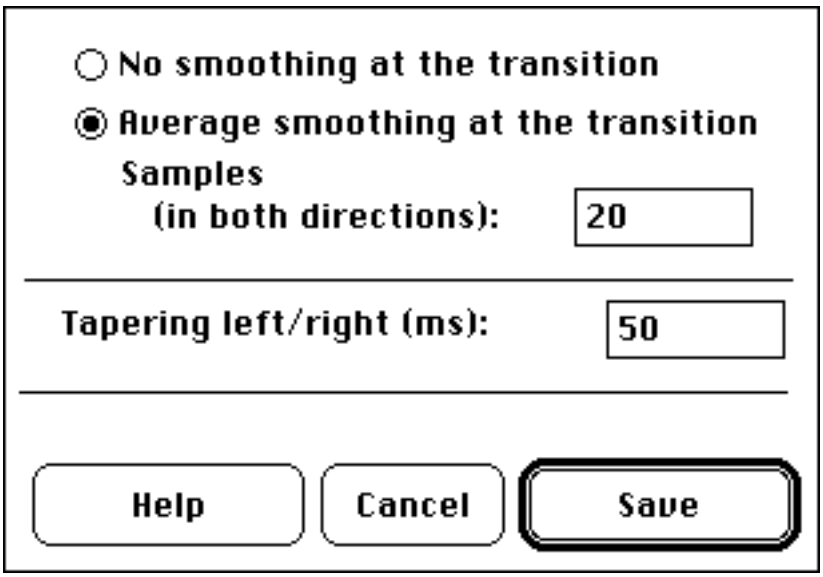
With **or** **(Text Edit)**
Standard clear text operation, operational in both the Edit and the Text Window. “Clearing text” means taking it out without saving it to a buffer.

Edit>Select All ⌘A
Edition>Sélectionner tout ⌘A
Bearbeiten>Alles auswählen ⌘A

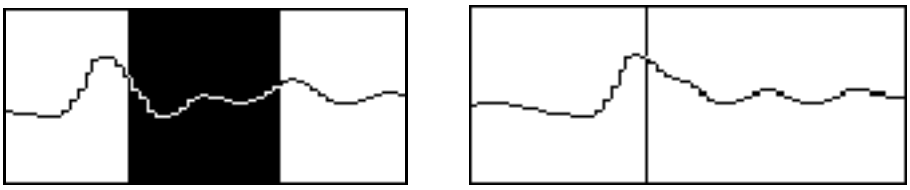
With **(Signal Edit)**
Selects all samples of the presently selected signal and blackens the signal. If you subsequently execute Edit>Clear, all samples are wiped out. (After such a wipe-out, an empty signal still retains header information, which occupy 8 bytes in the case of Signalyze and MacSpeech Lab files and information resources in the case of Sound Designer II, SoundEdit and SoundWave files.)

With **or** **(Text Edit)**
Selects all the text in the Edit or Text Window and blackens it. If you subsequently execute Edit>Clear, or press BACKSPACE, the text is wiped out.

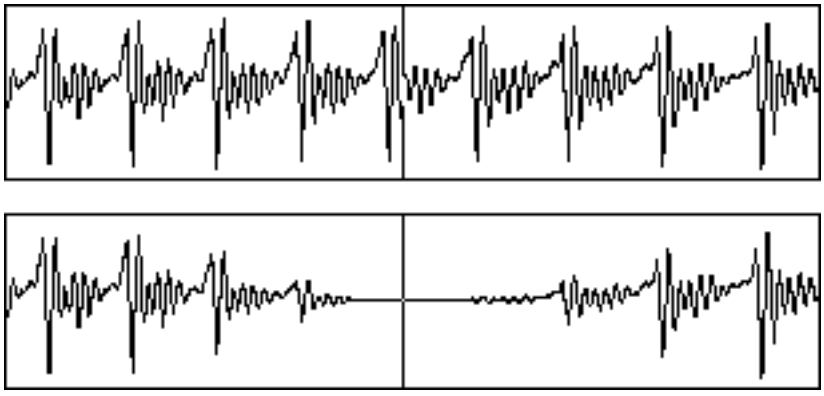
Edit>Signal Edit...
Edition>Edition du signal...
Bearbeiten>Optionen Signalbearbeitung...



- *Smoothing at the transition (signal splice):* Essentially, a straight line is drawn between the x^{th} sample on either side of the transition. In the example below, the blackened signal segment was removed and 3-point average smoothing was applied at the transition. Signal splices that respect the signal's Fo cycle often leave clickless new signals if relatively short average transition smoothing is chosen.



- *Tapering:* Specifies the distance over which tapering should extend to the left or to the right of the current cursor position. In the example below, a 30-ms window was specified for the right and left tapers. Tapering typically does not leave an audible click, but longer tapers can be heard as an amplitude dip.



- More detailed instructions for using smoothing and tapering are given in Tutorial 6 ("Signal Editing").

Edit>Taper Right / Edit>Taper Left

Edition>Fenêtrage gauche/Edition>Fenêtrage droit
Bearbeiten>Zuspitzen rechts/Zuspitzen links

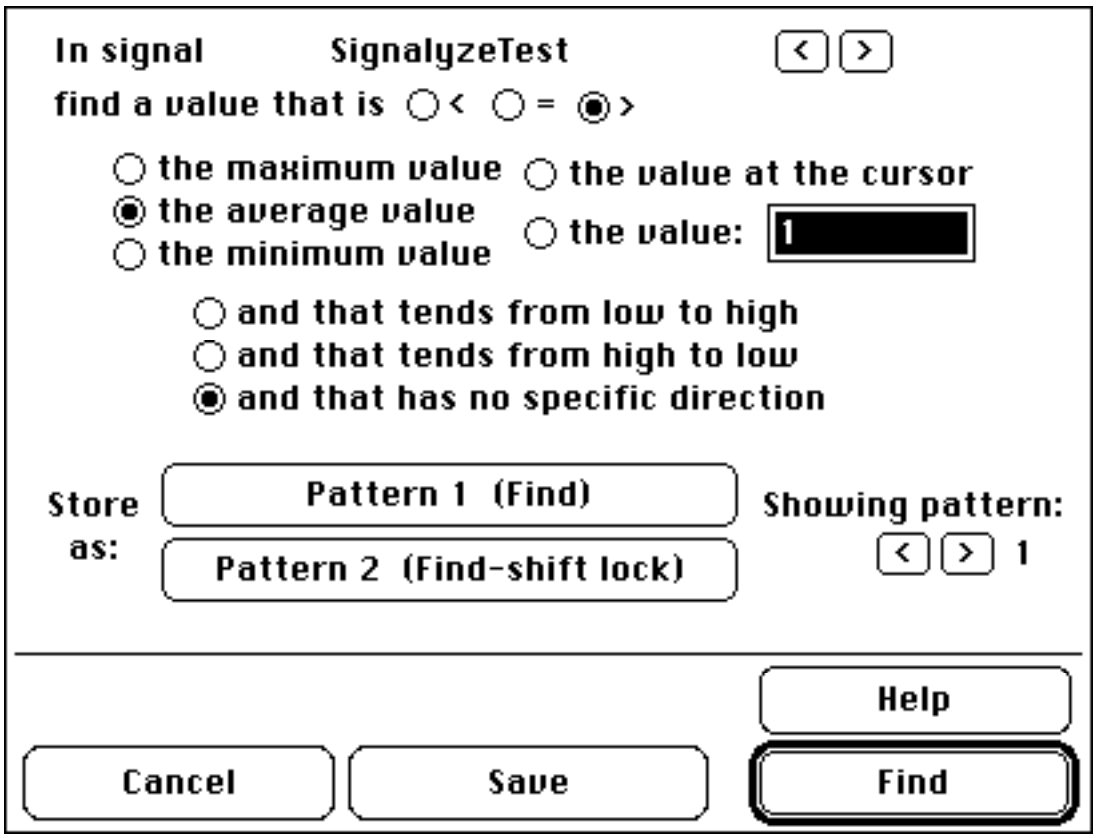
- *Taper right* means a cosine-defined signal taper, ending in a right-ended zero-deflection signal.
- *Taper left* means a cosine-defined signal taper ending in a left-ended zero-deflection signal.
- See previous entry and Tutorial 6 ("Signal Editing") for more details about tapering.

---

left cosine taper

Edit>Find in Signal... ⌘F
Edition>Recherche de signal... ⌘F
Bearbeiten>Signalsuche... ⌘F

(With the Signal Window in front)



Amplitude values greater than, less than, or equal to a given value can be searched for in a given signal. Two search patterns can be stored.

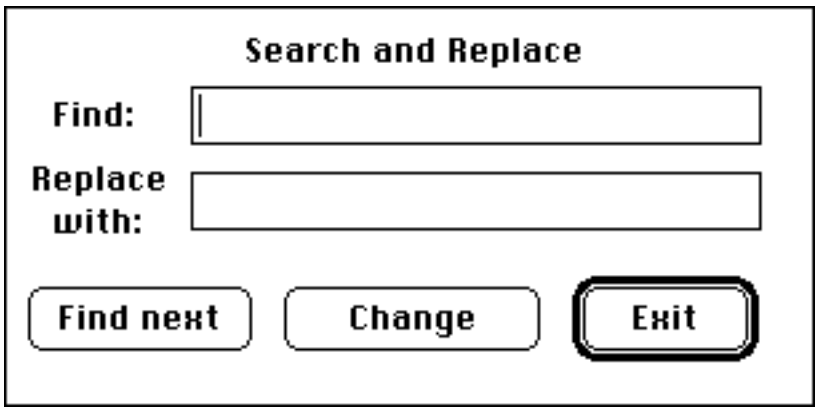
Follow the instructions in the Edit>Find in Signal menu command from top to bottom:

- Use the upper left/right arrows (< >) to select a signal.
- Set the logic comparison to “less than”, “equal to” or “greater than”.
- Select or write in the comparison value.
- Select the direction (if any) at the intersect between search value and examined sample stream.

- Store the pattern in one of the two pattern keys.
- Review the two patterns by with the lower left/right arrows ([<] [>]).
- Engage or disengage the SHIFT-LOCK key before you click on "Find" or "Save". Disengaged: pattern 1, engaged: pattern 2.
- After clicking "Find", the search begins at the cursor position. Results are reported in the Edit Window:
  — "The search value lies under the cursor." means either that the line cursor is placed right on the specified amplitude, or that its black outline covers values that *include* the specified amplitude.
  — "Value not found." means that the specified value does not lie to the right of the line cursor.
- Repeat the search by doing Edit>Find Again (COMMAND-E) after engaging/disengaging the SHIFT-LOCK key.
- It is often useful to "lock cursors" (Signal>Synchronous Cursors) to line up features in different tracks.

Edit>Text Search... ⌘F
Edition>Recherche de texte... ⌘F
Bearbeiten>Textsuche... ⌘F

(With the Text Window in front)



Performs search and replace operations in the Text Window.

- *To Find:* Write the search item into the **Find** box (capitalization is significant). Press "Find next".
- *To Replace:* Write the search item into the **Find** box and the substitution item into the **Replace** with box (capitalization is significant). Any string of visible characters is acceptable. Click on **Change** and the first occurrence of the search item is changed to into the substitution item.

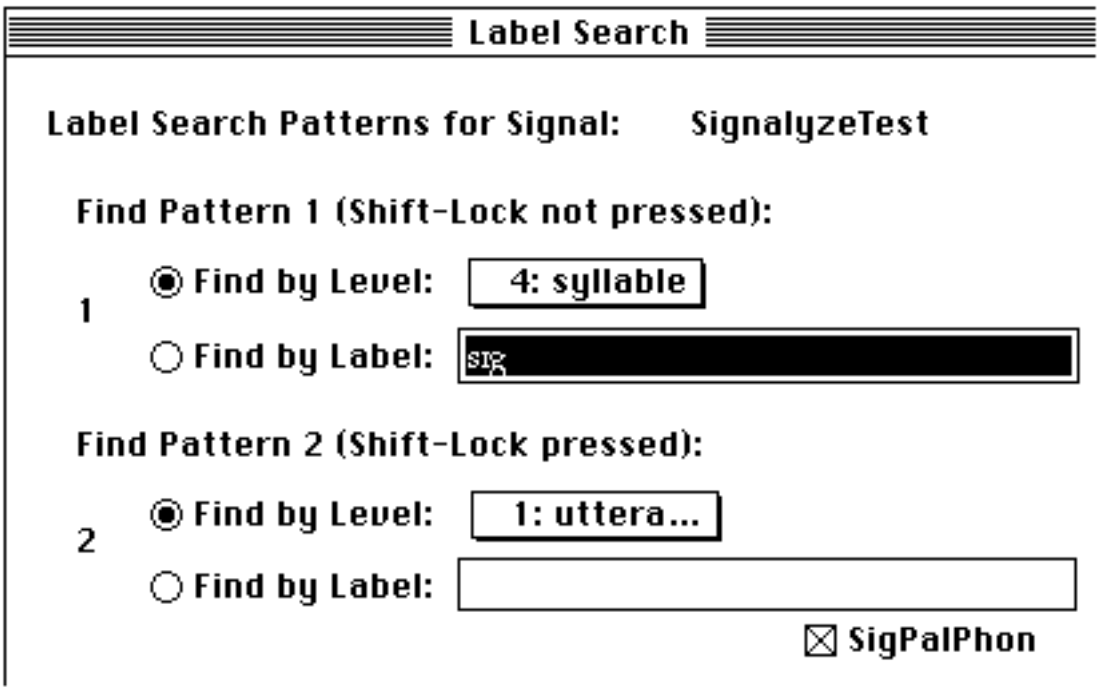
• *Continue:* You can find or change subsequent occurrences of the search item by continuing to click on the appropriate button.

Edit>Find in Signal Again ⌘E
Edition>Trouver à nouveau dans le signal ⌘E
Bearbeiten>Nochmals im Signal suchen ⌘E

(Only with Signal Window in front)

Repeats the find operation requested by Edit>Find in the selected signal. SHIFT LOCK key disengaged: find pattern 1, engaged: find pattern 2.

Search by Label...
Options de recherche par étiquette...
Optionen Etikettensuche...



You can search for labels using either a given labeling level or the label’s name. You can specify two separate search patterns. Switch between the two patterns with the SHIFT-LOCK key. The names in the level popup are those that are defined in the Edit>Label Setup dialog.

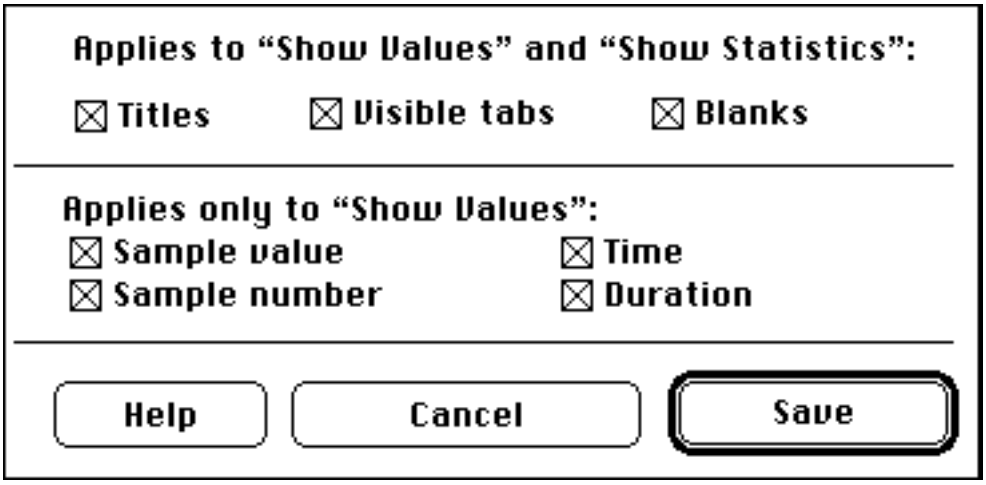
If you want to search for a label using the character set of a specific font (e.g., the phonetic font “SigPalPhon”) activate the font button at the

bottom of the Label Search dialog. The name of the button corresponds to the font you have selected for the signal through Edit>Label Setup.

Find Label Again ⌘L
Trouver étiquette à nouveau ⌘L
Nochmals Etikette suchen ⌘L

Once you've defined a search pattern through Edit>Search by Label, this menu command lets you search repeatedly for the same label pattern in a given signal.

Edit>Show Values Options...
Edition>Options pour affichage de valeurs...
Bearbeiten>Optionen Wertübertragung...



Specifies how a number is shown into the Edit Window.

Specifications in the Top Panel

- Controls the appearance of numbers called up through Edit>Show Values and Edit>Show Statistics:
- *Titles:* An appropriate *title* is automatically inserted according to the type of number you transfer. The table of title abbreviations is given in the appendix "Summary of Titles Used in Transfer Operations".
- *Visible Tabs*: When selected, invisible TAB characters appear as "¬" characters (hyphen-hooks) in the Edit and Text Windows. When not selected, TAB characters appear as blanks. Prior to each text save operation, the hyphen-hooks are converted into Tabs, and are reconverted to hyphen-hooks after completion of the operation.

• *Blanks:* Leading and trailing blanks surrounding a number can be suppressed if desired. This lets Signalyze show more values in the Edit Window and makes for shorter results files. On the other hand, the padding blanks let you line up columns in the text window with an equal-spaced font like Monaco or Courier.

Specifications in the Bottom Panel
Selects *which values* are transferred to the Edit Window by the Edit>Show Values function. Each of these options corresponds to an indicator in the Signal Control Panel (sample value = bottom number, sample number = top number) or in the General Control Panel (time = "s/ms/μs", duration = "d").

Edit>Show Values ⌘G
Edition>Afficher valeurs ⌘G
Bearbeiten>Werte übertragen ⌘G

Transfers to the Edit Window those of the four current numeric indications (sample number, amplitude, time indication and duration indication) that have been specified in Edit>Show Values Options.

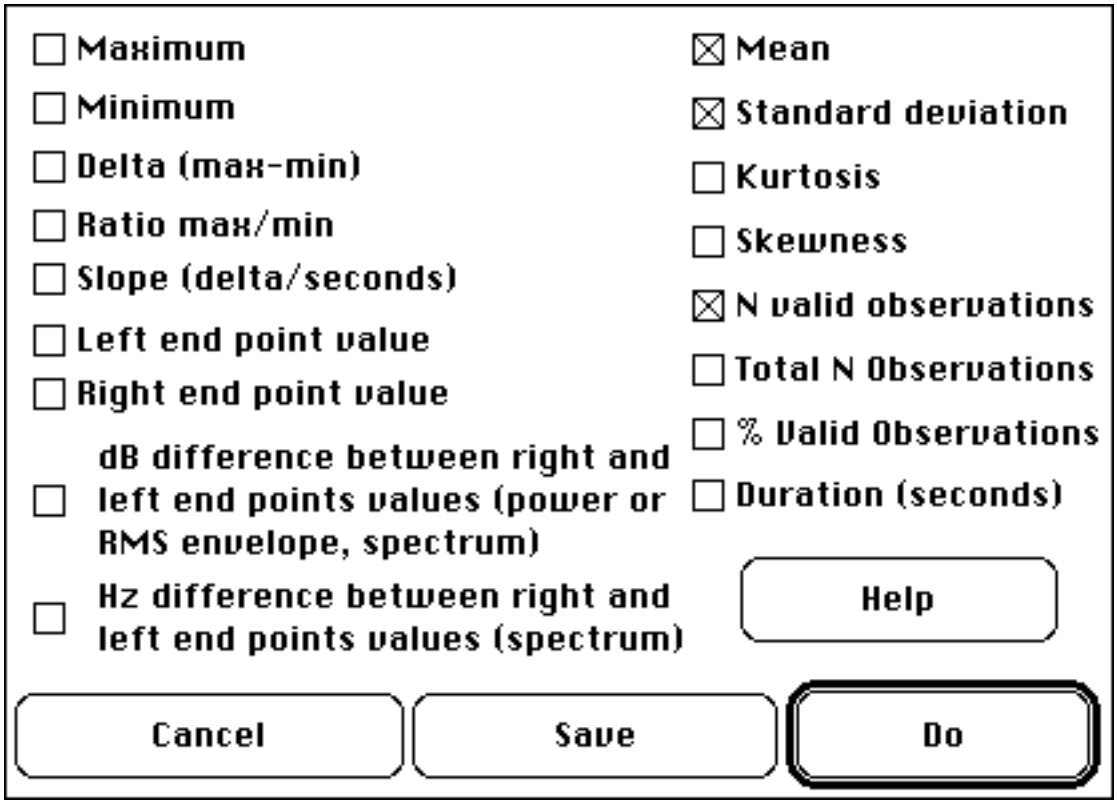
• Place the cursor in the place of interest and activate this command. The specified values are shown in the Edit Window. Do Edit>Send to Text Window (COMMAND-J) or press RETURN to transfer values to Text Window.

• Titles, Tabs and blanks are according to specifications in Edit>Show Values Options.

• Numeric format is according to specifications in Signal>Numeric Format.

Edit>Show Statistics Options...

Edition>Options pour affichage de statistiques...
Bearbeiten>Optionen Statistikübertragung...



Specifies statistics to be performed on the *selected (blackened) segment* of the currently active signal.

- Internally, sample sums are calculated in *32-bit integer* registers to save time, but statistical calculations are performed in *floating point*. Because of the 32-bit sample sums, the routine may fail for very *long signals* (e.g., above 50,000 samples).
- *Basic Operation:*
  — Select statistics in “Show Statistics Options”
  — Specify display format with “Show Values Options”
  — Blacken a portion of a signal or a spectrum
  — Activate “Show Statistics”
  — Statistics are shown in Edit Window
  — Transfer statistics to Text Window with RETURN
- *Note on “missing values ”:* Statistics operations apply only to valid samples, and not to “missing values”. “Missing values” are non-displayed samples, such as those that make up the unvoiced and thus blank segments of a pitch extraction. “Missing values” are internally represented with the value +32767.

- *Minimum, Maximum, Delta, Ratio, Slope:* These operations relate to the minimum and maximum values in a given signal segment. "Missing values" are not counted.
  - — Minimum, maximum: Minimum and maximum values in selected signal segment.
  - — Delta: Arithmetic difference between *minimum* and *maximum*.
    CAUTION: This is rarely equivalent to the difference between the *two end points* of a signal selection.
  - — Ratio: The result of the division of the maximum value by the minimum value.
  - — Slope: Delta/duration, given as (maximum-minimum)/seconds
- *Left and Right End Point Values, dB and Hz Differences:* These operations relate to values found at the two extremes of a signal selection. If one or both of these is a "missing value", the requested operation is not performed.
  - — Left and Right End Point Values: The first values inside the selected (blackened) portion of a signal.
  - — dB Difference. The amplitude difference between the two end points, expressed as dB. This operation applies only to power and RMS envelopes and to spectra.
  - — Hz Difference. The difference in Hertz between the two end points. This operation applies only to spectra.

How dB Differences are Calculated in Spectra and Envelopes

dB differences in spectra are calculated differently from those in amplitude envelopes, but they are always calculated on the basis of end point values. In cases where it is better to take the raw end point values, calculate dB differences like this:

Spectra:

```
left dB = 20 * log10(left end point value)
right dB = 20 * log10(right end point value)
difference dB = left dB - right dB
```

Envelopes:

```
predecibels = log10(left endpoint value/
                                right endpoint value);
if (envelope==power) difference dB = 10 * predecibels;
else if (envelope==RMS) difference dB = 20 * predecibels;
```

- *Mean, sd, kurtosis, skewness:* Standard statistics operations applying to all non-missing values in a selected signal segment.
  - — Mean, sd: Measures of average tendency and degree of variation.
  - — Kurtosis, skewness: Measures of deviation from a normal distribution, whereby kurtosis measures degree of "vertical" peakedness/flatness and skewness measures "horizontal" deviation from a centered peak.

- *Number of Observations:* These measures report how many values constitute the currently selected signal segment.
  - — N Valid Observations: Number of values other than "missing values".
  - — Total N Observations: Total number of observations, including "missing values".
  - — % Valid Observations. Percent of total values in selected signal segment that are not "missing values".
- *Duration:* The duration of the selected signal segment, measured in seconds.

Edit>Show Statistics ⌘U
Edition>Afficher statistiques ⌘U
Bearbeiten>Statistiken übertragen ⌘U

Transfers those of the current statistical indications to the Edit Window that have been specified in Edit>Show Statistics Options.

- Select (blacken) the part of the signal you want to analyze.
- **If you don't blacken any specific part of the signal, statistical values will be calculated on the basis of the entire visible portion of the signal.**
- Activate this command. The specified statistics are calculated and placed in the Edit Window.
- Place the cursor in the place of interest and activate this command. The specified values are shown in the Edit Window. Do Edit>Send to Text Window (COMMAND-J) or press RETURN to transfer values to Text Window.
- Titles, Tabs and blanks are according to specifications in Edit>Show Values Options.
- Numeric format is according to specifications in Signal>Numeric Format.

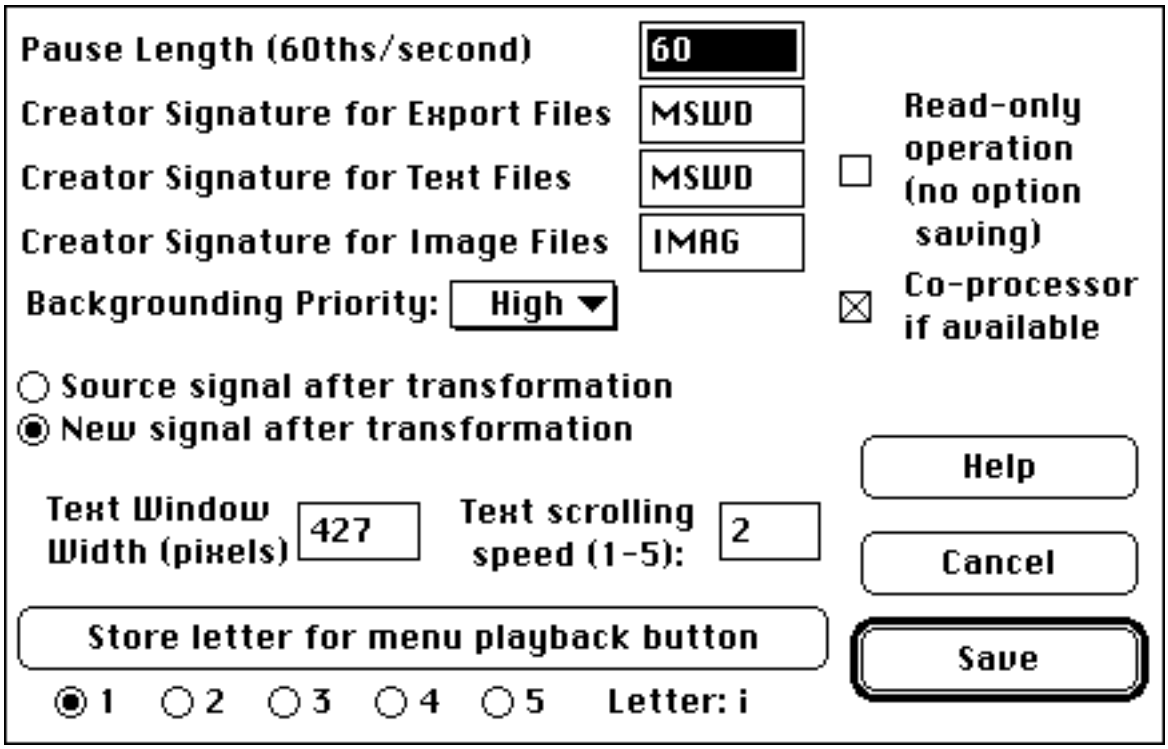
Edit>Send to Text Window ⌘J
Edition>Envoyer à la fenêtre de texte ⌘J
Bearbeiten>Ins Textfenster senden ⌘J

Transfers contents of Edit Window to the Text Window.

- *Short cuts:* Press RETURN or double-click in signal.

Edit>Preferences...

Edition>Préférences...
Bearbeiten>Wahleinstellungen...



This dialog specifies a number of personal preferences.

- ***Pause Length:*** Some messages are displayed just long enough to let the user know what's going on. Example: In a dialog box, typing a letter where a number is requested causes the message to be flashed: "Use only numbers." Set the duration of such displays in sixtieths of seconds. Minimum: 0, maximum: 600 (6 secs).
- ***Creator Signatures:*** Each Macintosh file is marked by two types of identification, the *file type* and the *creator signature*. The *file type* distinguishes for example signal, text and image files. The *creator signature* is associated with specific applications. For example, a text file saved with Microsoft Word bears the signature MSWD, while a compatible and otherwise indistinguishable text file saved with Excel bears the signature XCEL. This lets the Finder locate the correct program when the user double-clicks on the document.

By specifying the creator signature, an efficient data transfer between Signalyze and other applications can be planned. E.g., if manual scoring results are destined for Word, specify MSWD as the "Creator Signature for Text Files". All files saved from the Text Window will be saved with that signature. After leaving Signalyze, double-clicking on the documents loads the file into Microsoft Word (if available). Also, the document often "wears" the icon of the destination application. Here are some common creator signatures for text files:

MacWrite	MACA
Microsoft Excel	XCEL
Microsoft Word	MSWD
StatView	STAT
Edit	EDIT

Note: "Export File" in this dialog means "*signal* file saved in ASCII/TEXT format".

• ***Background Priority:*** Signalyze can operate in background, while you're working in another application. "Background Priority" lets you set the importance of Signalyze processing in the background:

— *High:* Signalyze "hogs" the processor. It tries to gain as much of its time as possible. As a result, other applications running in the foreground may be sluggish, even on faster computers.

— *Middle:* Signalyze takes its fair share of background processing time, but doesn't "hog" the processor. On many mid-sized Macintoshes, this may be a fair setting.

— *Low:* Signalyze profits primarily from periods of low processor demand.

• ***Co-Processor Use:*** To speed up some processes, such as signal filtering with the Butterworth or Chebychev filters, Signalyze jumps to special external routines that use the mathematical co-processor, if it is available.

— The *disadvantage* of using such external routines is that to some degree, the program "loses touch" with the user and with the main program. For example, the usual backgrounding priorities are suspended, because in co-processor mode, the routine takes all available processor time. Also, the usual progress indicator is unavailable in co-processor mode, and continued operation is signaled through a blinking cursor. Finally, clicking in the Signal Window is no longer sensed, and the process cannot be interrupted by clicking on the Abort button. However, the process can be interrupted with COMMAND-period.

— Co-processor use can be turned on or off through the "Use co-processor if available" option.

• ***Read-only Operation:*** This option indicates if R/O mode is in operation and turns read-only mode on and off. Read-only operation only affects the saving of session settings and not the saving of files.

— If Signalyze is run from a locked disk (software- or hardware-locked), the program puts itself *automatically* into read-only (R/O) mode. Many installations prefer to keep their hard disks locked, in order to prevent exposure to viruses.

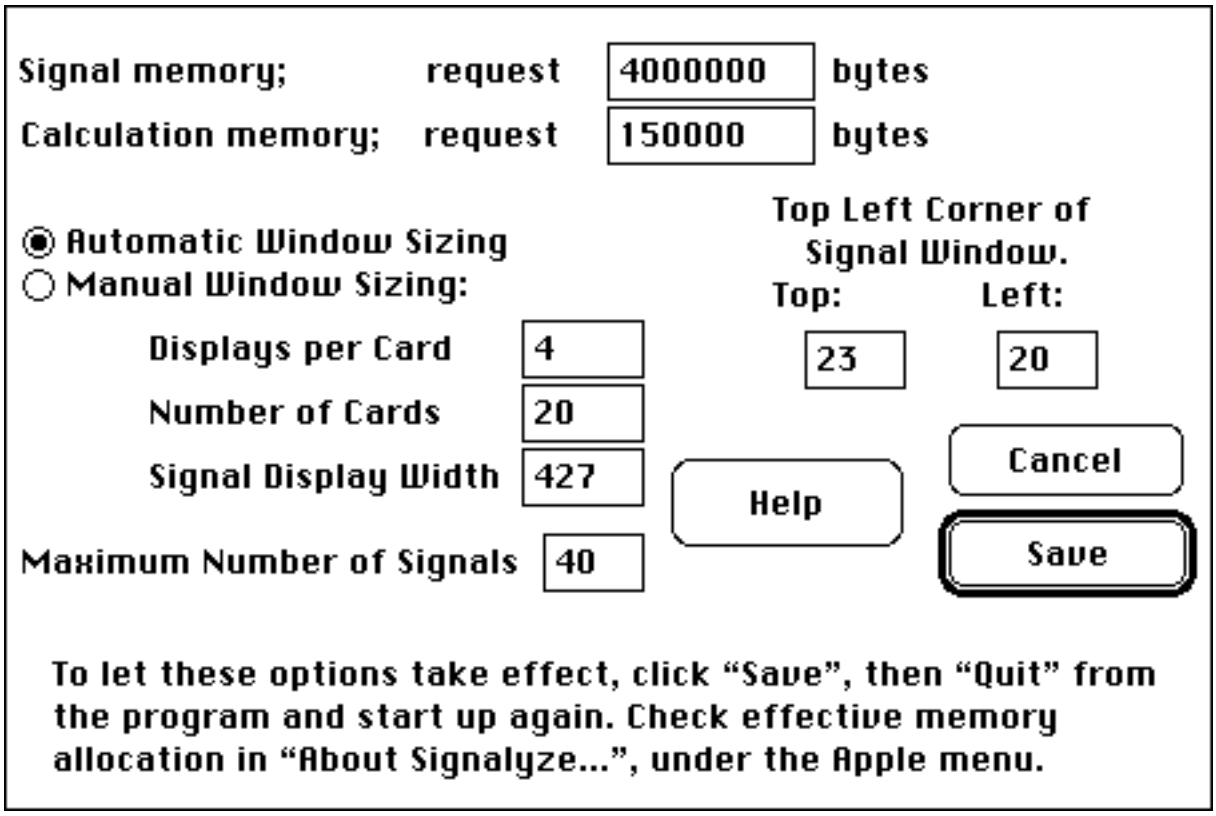
— To initiate Signalyze in R/O mode from a disk that is *not* locked, hold down both OPTION and COMMAND keys shortly after starting up the program.

— A side benefit of R/O mode is that the program exits faster.

- ***Source signal/New signal after transformation:*** This option specifies which signal becomes the selected (active) signal after a transformation (e.g., pitch extraction, arithmetic transformation, spline, etc.) is performed. The *source signal* option is useful when performing several transformations on the same source signal, while the *new signal* option is preferable when performing chains of transformations.
- ***Text Window Width:*** "Text Window Width" specifies after how many pixels carriage returns are inserted into Text Window lines. In spreadsheet fashion, the Text Window width can be set wider than available screen size so that long rows of numbers align correctly. For technical reasons, the Text Window width cannot be set smaller than screen width.
- ***Text Scrolling Speed:*** COMMAND-down/up arrow lets you scroll gently through the text window. "Text Scrolling Speed" specifies the speed at which the scroll occurs (1=slowest, 5=fastest)
- ***Store Key for Menu Playback Button:*** This facility lets you store any *visible letter* as a short cut for calling a menu playback button in the General Control Panel. To store a letter, press the key and then furnish a letter. The stored letter is shown for each menu playback button.

Edit>Startup Options...

Edition>Options de démarrage...
Bearbeiten>Starteinstellung...



Dialog to configure Signalyze memory requirements and display characteristics. With these options, the program can be adapted to your specific needs, machine, or display monitor. You can make full use of available memory or limit Signalyze to a portion of it. You can set up the displays so that they take less monitor space, or you can let Signalyze fit the displays to the full size of your (main) monitor.

- The specifications on this dialog box take effect when you *restart Signalyze*. After "Saving" the changes, "Quit" the program and start up again.
- *Effective memory allocations* can be verified in the >About Signalyze menu.
- See the Technical Information section for more details on memory management, particularly with respect to the difference between calculation and signal memory.

Memory Under System 7

Under System 7, there are **two place**s to set memory:

1. Through the File>Get Info box at the desktop level.
2. In the Edit>Startup Options dialog box.

- At the desktop level, set the *overall* amount of memory. This includes: The Signalyze application, calculation memory, signal memory and any extra memory reserved for print drivers. Initially, Signalyze 2.0 is set to 2000 k minimum and 4000 k preferred. This can be increased safely to about 500 k less than maximum available RAM.
- In this dialog box, set the memory used by *Signalyze*. In contrast to operations under Finder in System 6, you do *not* have to allow for large desk accessories (they're in a separate layer). However, you *do* have to worry about large print drivers potentially eating into your Calculation Memory.

More on Memory

Signalyze maintains two types of memory, calculation memory and signal memory.

- ***Calculation Memory***: Most calculations performed by Signalyze require no more than 300000 bytes (300 k). But to be on the safe side, it is good practice to set calculation memory higher than 300 k. If you're strapped for memory, you can always experiment with reducing the calculation memory to as little as 200 k.

• ***Signal Memory***: This is memory reserved for signals and their transforms. The request for signal memory can be as high as desired. Upon startup, Signalyze figures out how much memory actually is available *after* allowing for calculation memory, and as much effective signal memory as available is attributed.

More on Window Sizing, Cards and Placement

• If the *"Automatic Window Sizing"* option is active, Signalyze configures the signal display to nearly fill the screen. The Text Window is always sized the way it was left in the last session, even if it was configured to show on a second monitor. Either window can be dragged. The Text Window can also be re-sized. Set place of carriage return insertion for the Text Window through Edit>Preferences.

— *Handy placement of Text Window:* If the Text Window is displaced slightly to the left of the Signal Window, you can easily click back and forth between the two windows.

• With the *"Manual Window Sizing "* option, the number of displays per card, the number of available cards, the signal display width (in pixels) and the placement of the top left corner of the signal window (in pixels) can be specified. The top left corner specification is handy for users in Japan whose menu bar is taller than the standard bar.

• *Permissible specifications:* Displays per card: 4-40; number of cards: 2-100; signal display width: 427-4000; top of Signal Window: 22-4000; left edge of Signal Window: 2-4000.

• *Maximum Number of Signals:* There is a small amount of memory overhead associated with each signal space. It is best to define the maximum number of signals so as to retain enough flexibility without incurring too much overhead. Remember that each transformation or pitch extraction takes up another signal space. Permissible: 8-100 signals.

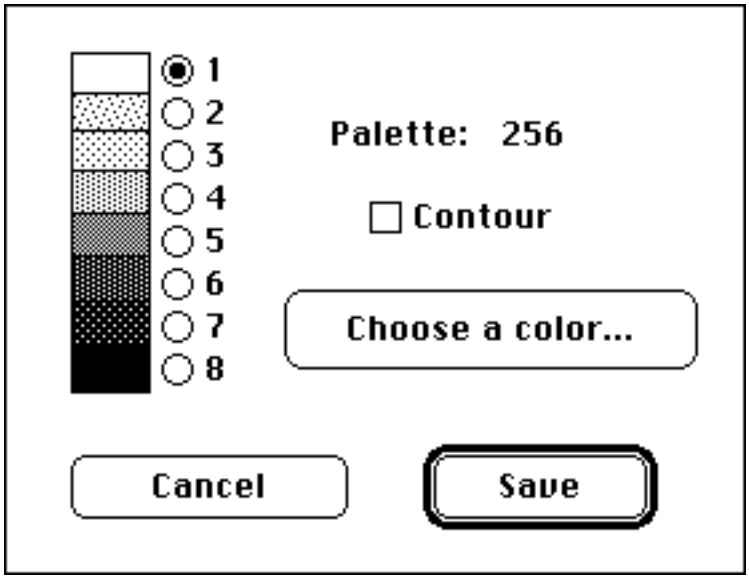
Edit>Language...

Edition>Langue...
Bearbeiten>Sprache...

Dialog to change Signalyze's user interface language.

Edit>Spectrogram Palette...

Edition>Palette spectrographique...
Bearbeiten>Spektrographische Palette...



This menu is only active on computers capable of high resolution (gray scale and color), and with monitors *set to 16 colors or higher*. The dialog lets you define the color scheme used in drawing spectrograms. The available spectrum (16/256 grays or colors) is subdivided into eight settable color domains, and the remaining shadings are calculated by interpolation.

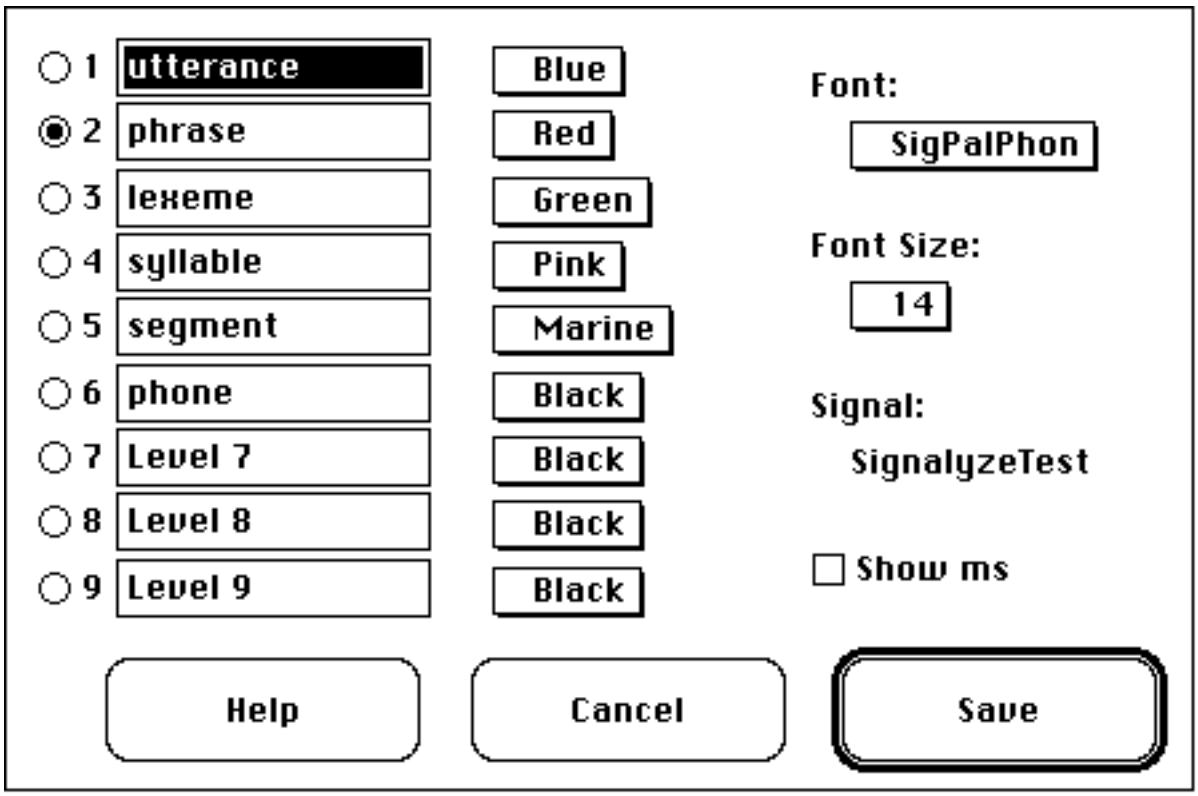
- There are *four separate palettes* for 16 grays, 256 grays, 16 colors and 256 colors. For display devices with more than 256 grays or colors, the 256-color palette is used.
- *To redistribute colors/grays:* For each gray scale or color domain (1-8), click the appropriate button and activate the button “Choose a color”. In the standard color picker dialog, choose a color and adjust brightness with the scroll bar. The previous color is shown as the lower half, and the new color as the upper half of the upper left rectangle. Exit with “Ok” when satisfied.
- If working with grays, click in the middle of the color wheel and set the brightness with the scroll bar.
- *Contour (only visible with 256-color/gray option):* The optional contour effect is created by calculating colors as “inclined steps”.
- If you often use *“Inverse Colors ”,* redefine your color palette from dark to light, rather than from light to dark. Running the spectrogram routine without “Inverse Colors” option saves time.
- *Contrast Setting:* If the *spectrogram seems too pale* for your taste, emphasize the lighter colors/grays at one end, the darker colors/grays at the other end, and let the middle change more abruptly from light to

dark. If the spectrogram seems *too stark,* reduce contrast by inverting these tendencies.

Label Setup...

Options d'étiquetage...
Etiketteneinstellung...

(For more details, please consult Tutorial 5)



This dialog sets a number of parameters used in labeling operations.

- The nine levels specify (not necessarily hierarchical) levels used to group labels. Each label has necessarily a level, such as "segment", "syllable", "phrase" etc. Each level has its own label color (if you have a color monitor). Labels are marked for their level by a number placed in front of the label name (e.g., "4: sig", means that the label "sig" is marked for the fourth level). Label levels can be used in label searches and in post processing of label information.
- Specific font for labels are chosen with the popup menu in the upper right corner. **Suggestion:** If you want to use a phonetic font for labeling, stick to *SigPalFont* (the shareware font supplied with Signalyze). SigPalFont preserves the numbers and angular brackets you need to indicate labeling levels, which is usually not possible with other phonetic fonts.

- If you choose the option “Show ms”, the exact millisecond position is shown alongside with the label.
- The active level is indicated by the button on the left edge of the dialog. Outside this dialog, you can switch between labels with a popup menu obtained by clicking in the signal with CONTROL-shift.
- **Label information is stored as “TEXT” files. The structure of label files in indicated in the “Technical Information” section of this manual, under “Label Format”.**

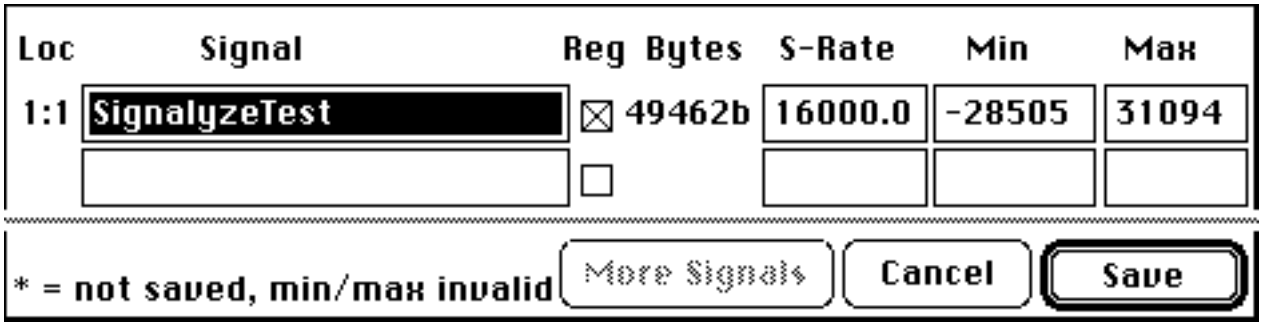
Show Labels

Afficher les étiquettes
Etiketten zeigen

Menu command to switch the label display on or off for signals that have labels.

“Signal” Menus

Signal>Signal Information... ⌘I
Signal>Informations sur signaux... ⌘I
Signal>Signalinformation... ⌘I



This dialog provides summary information about the signals currently stored in the program. When more signals are stored than are shown, a click on the button “More Signals” brings up another “page” of information.

- The information under *“Loc ”* specifies the *location* of the last version of the signal. If only one display exists, it specifies the display where it was first loaded in. If several displays of the same signal exist, the last-created display is listed. If no current display exists for a signal in memory, no location is indicated. The first number specifies the card and the second number specifies the display.
- *“Reg ”* means “ regular signal” if checked, or “derived signal” if unchecked. Regular signals are “originating” signals, derived signals are those that depend partially on another signal (e.g., spectra, pitch extractions).

— To change a derived signal into a regular signal, enable the check box marked “Reg” next to the signal in question. This opens derived signals to a number of additional processes, such as filtering and Find operations.

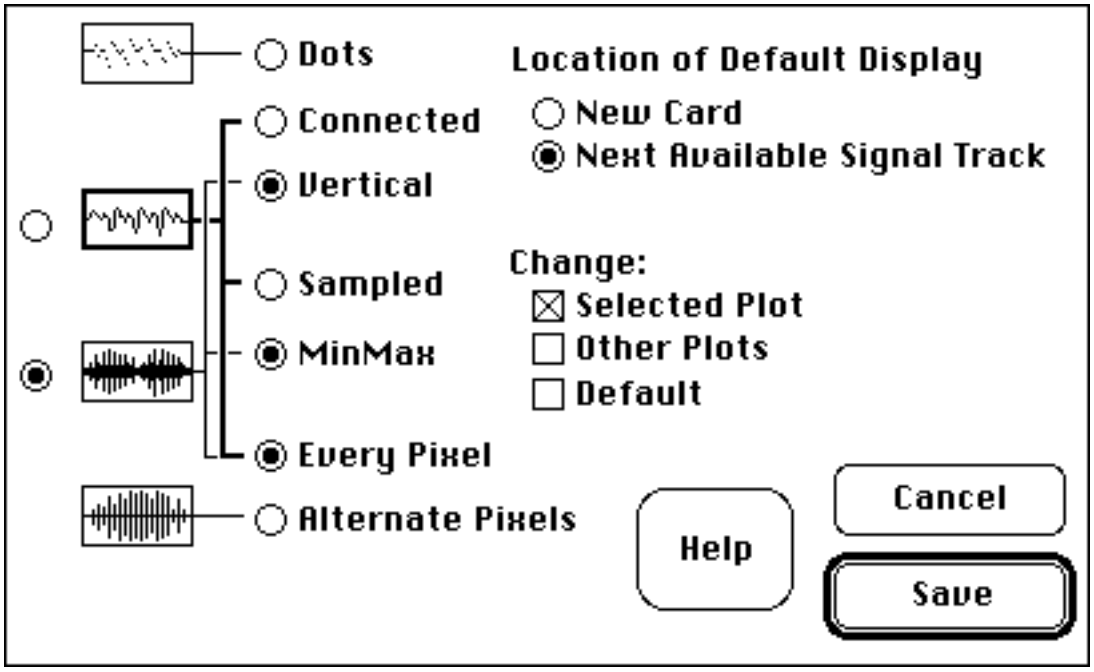
— A regular signal cannot be changed back into a derived signal, since a number of parameters are irrevocably changed by the Derived —> Regular change.

- The *file length* is given in bytes. To calculate the length in samples, divide the byte length by 2.
- The *sampling rate* is given in Hertz. This rate can be modified by the user. With sound signals, this has the effect of raising or lowering the overall pitch of the signal.

• *Minimum and maximum values* control the upper and lower bounds of the signal’s display. These values cannot exceed ±32,767. They can be modified by the user to permanently modify the visual representation. (For a faster way of setting the minimum and maximum values, see the Signal>Set MinMax command, below).

Signal>Display Setup...

Signal>Options d’affichage...
Signal>Optionen Signaldarstellung...



PLOTTING
Three variables are manipulated to display signals in different ways:
• 1. Pixels drawn on the y-axis (dots, connected lines, vertical lines)
• 2. Samples examined (sampled, MinMax)
• 3. Pixels drawn on the x-axis (every pixel, alternate pixels)

• ***Variable 2: Samples Examined:*** It’s best to begin with variable 2. To obtain a *sampled* display, *one* sample is chosen for each vertical pixel line making up the plot. Taking into account sampling frequency and zoom information, samples lying *closest to the left edge* of each vertical pixel line are used as input for the display.

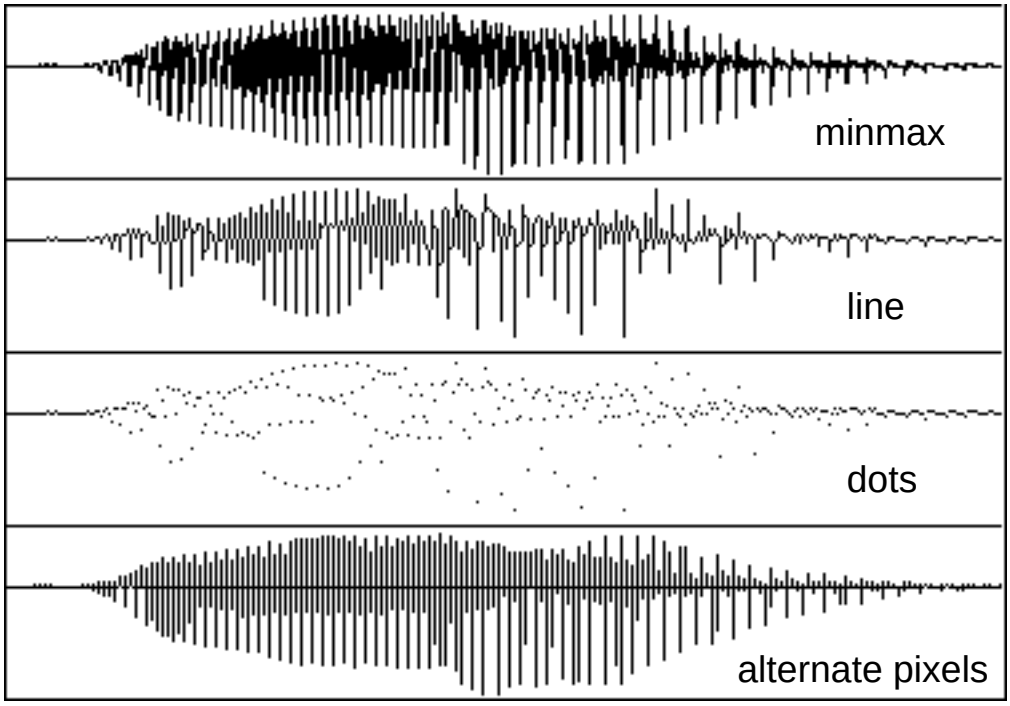
To obtain a *MinMax* display, *all* samples “lying under” the display pixel are examined, and the least and greatest value are used as input for the display algorithm.

- ***Variable 1: Pixels drawn on y-axis:*** Once the input values are known, the program has a choice over how many pixels are used on the y-axis. For the *dot plot*, a single value is used. If the "sampled" option is in effect, that value is the chosen sample. If the "MinMax" option is active, that value is the maximum value. It is usually best to chose the "sampled" option when a dot plot is desired, because otherwise, the display turns into a type of amplitude envelope. However, the choice is up to the user.

The *connected line plot* connects the previous value to the present value with a line. If the previous value is lower than the present value, the line rises from the previous value to the present value; if the reverse is the case, the line falls from the previous value to the present value. Again, use the "sampled" option with this plot, since connecting the maxima through MinMax produces an amplitude envelope.

The *vertical line plot* draws vertical lines from the minimum to the maximum value, plus extensions to connect the MinMax line to the previous MinMax line. This is the natural way of representing "MinMax" values; in dense to semi-dense displays, this often provides a maximally informative time/amplitude display for raw acoustic signals.

- ***Variable 3: Pixels drawn on x-axis:*** The final variable controls whether every pixel or alternate pixels are drawn. If *every pixel* is drawn, calculations are repeated for each pixel line. If *alternate pixels* are drawn, calculations are performed only every other pixel. The signal's average value is displayed on in-between lying pixels.
- ***Combination Buttons:*** To produce particularly effective displays, two combination buttons are available at the extreme left of the dialog box. As the illustration indicates, the top button activates the combination Connected-Sampled-Every Pixel, while the bottom button activates the Vertical-MinMax-Every Pixel combination.

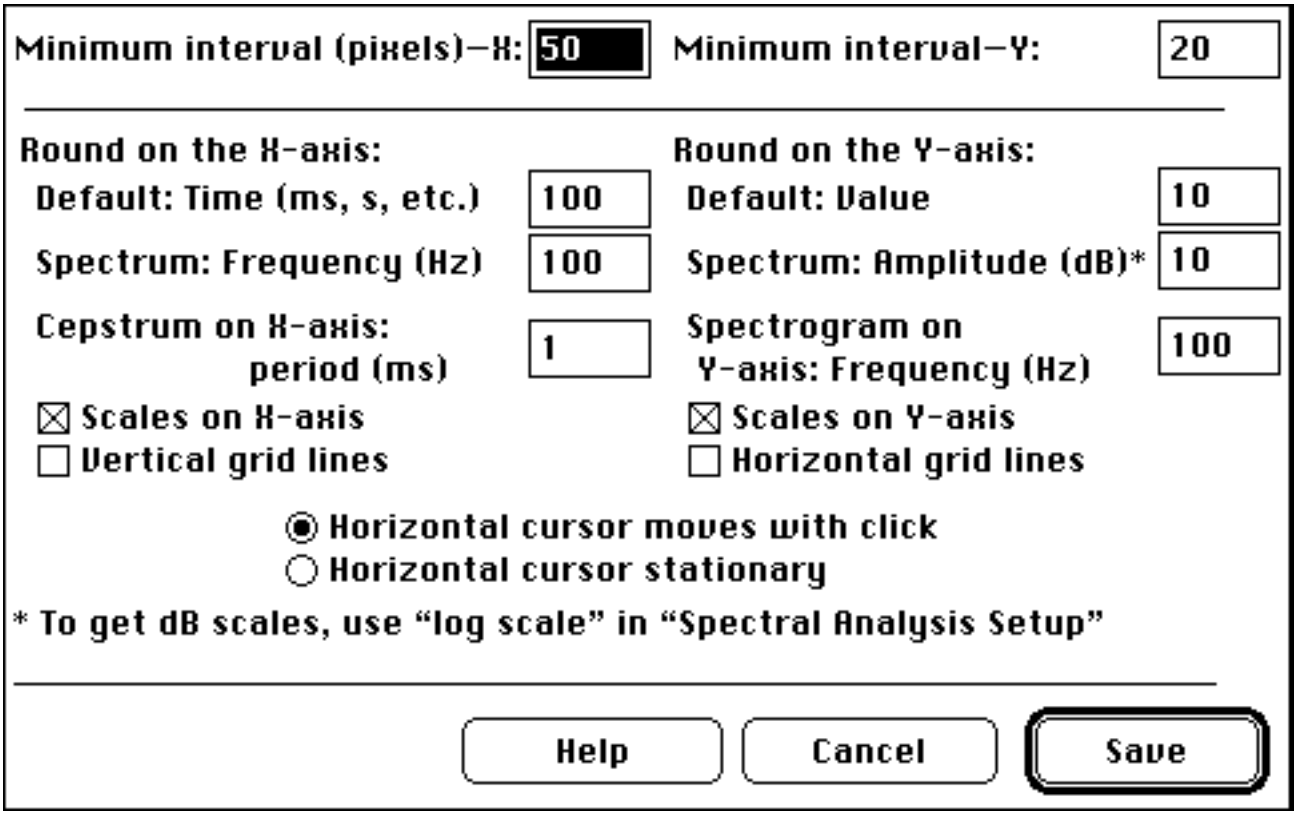


DEFAULT AND CURRENT SETTINGS

- Upon entry into the dialog, the user sees:
  — The setting of the *currently selected* signal if there is any.
  — The present *default plot setting* if there is no currently selected signal. This is the setting that will be applied to new signals. This may be different from the default setting for *derived signals* which is always the “connected line” option. For most derived signals, other plots wouldn’t look right.
- *Change selected plot:* Applies the specified changes to the currently selected plot.
- *Change other plots:* Applies the specified changes to plots other than the selected plot.
- *Change default:* Applies the specified changes to all future plots. Use this option if you want to change the *default plot settings for new signals.*
- *Location New Card:* New signals are directed to the first card containing an empty first display slot.
- *Location Next Available Slot:* New signals are directed to the first empty display slot, no matter on which card it may be found. A selected default display location can be *overridden* by the use of the Destination Button on the Signal Control Panel.

Signal>Scale Setup...

Signal>Options d'échelle...
Signal>Optionen Maßstab...



This dialog controls the display of numeric scales, separately for x or y, as well as the insertion of horizontal and vertical grid lines. Also according the options of this dialog, the horizontal cursor can either follow mouse clicks or remain stationary.

- ***How Scales and Gridlines are Positioned:*** Scale numbers are drawn when two criteria are satisfied:

  — the new number is at least a specified number of pixels removed from the previous number ("minimum interval").

  — the new number can be divided by the specified rounding number ("rounding").

  Gridlines are positioned in analog fashion. Fill in rounding numbers according to type of plot. Remember that you can see dB scales in spectra only if you also select "log scale" in "Spectral Analysis Setup".

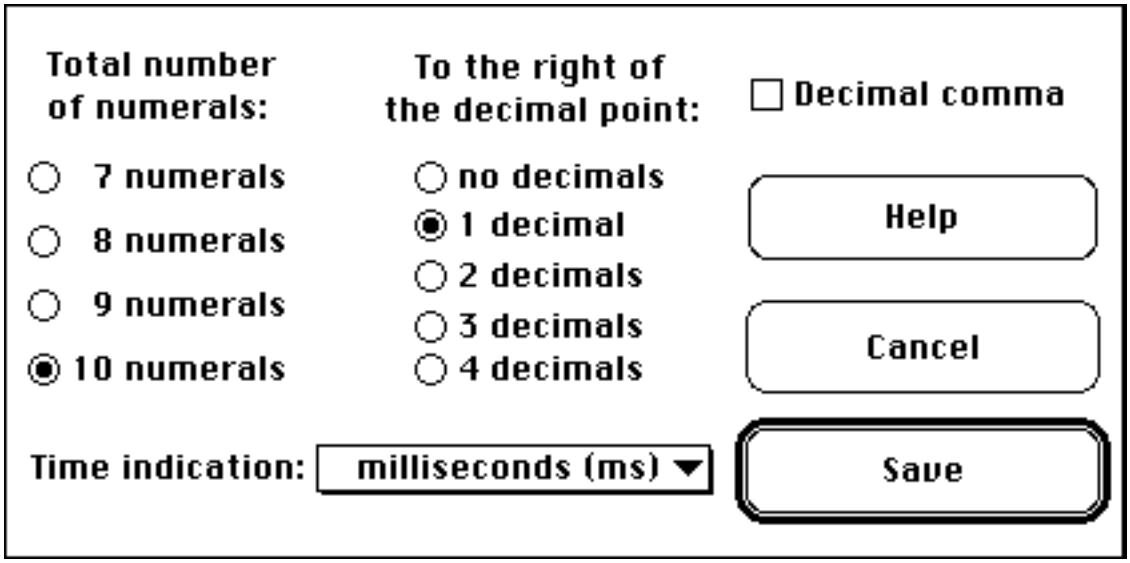
- ***Horizontal Line***

  — "At the click": The horizontal line cursor jumps to the click.

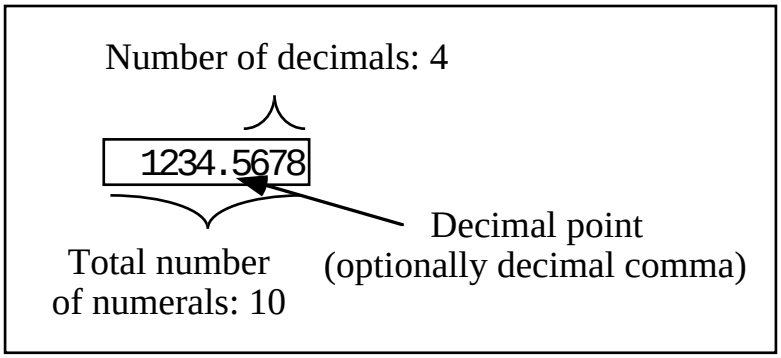
  — "Stationary": The horizontal line cursor remains stationary.

  — Toggle between "At the click" and "stationary":
  OPTION-click on horizontal line button: "At the click".
  COMMAND-click on horizontal line button: "Stationary".

Signal>Numeric Setup...
Signal>Options numériques...
Signal>Optionen Numerische Formate...



The Signal>Numeric Setup sets the form in which decimal numbers are shown.

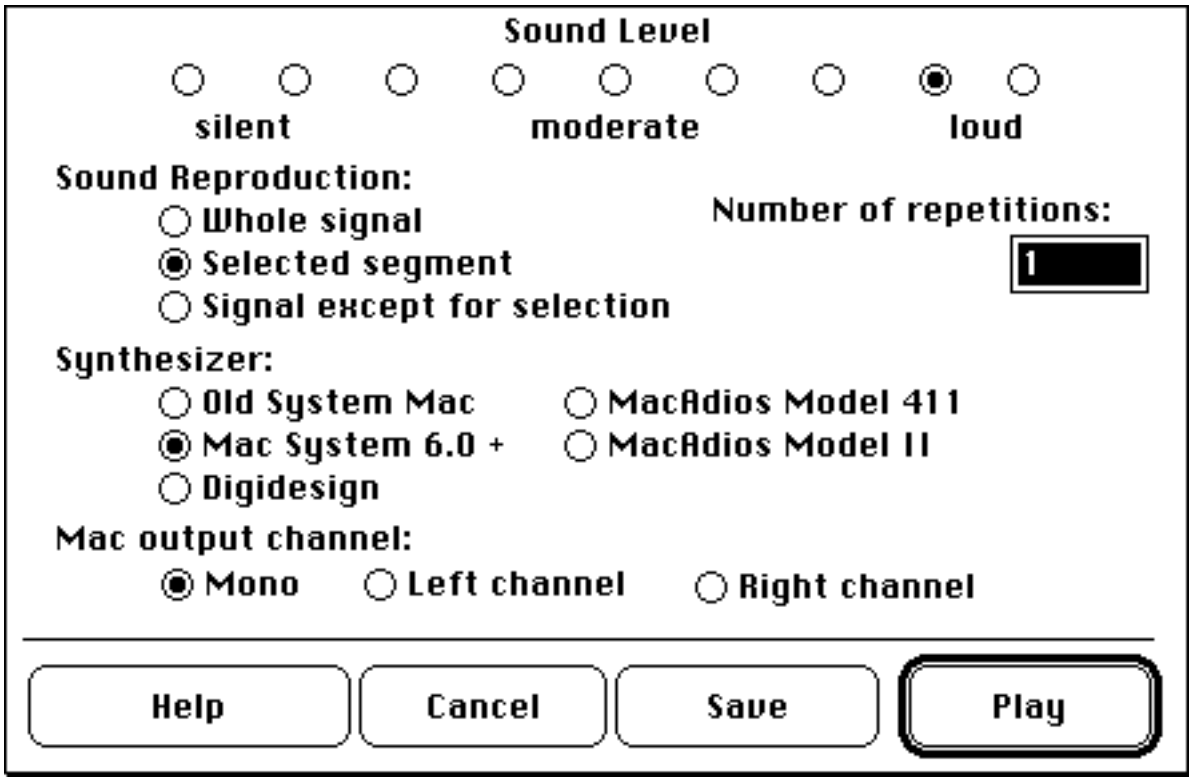


The dialog controls a number of aspects of the appearance of numerals:

- ***Total number of numerals shown:*** Number of numerals making up the number. If "blanks" is selected in Edit>Show Values Options, unoccupied numerals are filled with blanks.
- ***Number of decimals:*** The numerals to the right of the decimal point or decimal comma.
- ***Decimal point/comma:*** The symbol used to separate the integer and the decimal portions of the number. People outside English-speaking areas of the world may wish to select the decimal comma option. However, remember that most current spreadsheet and statistics programs do not support the decimal comma standard.
- ***Time indications:*** in seconds, milliseconds or microseconds.

Signal>Audio Setup...

Signal>Options pour audio...
Signal>Toneinstellung...



This dialog box sets the playback options.

- ***Built-in Synthesizers:*** During startup, the built-in synthesizers are selected according to your processor and System version. Computers with the M68000 and/or using a System prior to System 6.0 are switched to the "old-style" audio drivers (exception: Portables and PowerBooks, since their clock cycle is faster). Computers with faster processors running System 6.0 or later are switched to the "new-style" Sound Manager calls which provide a softer, more realistic tone. "Standard" Macs with M68000 processors (MacPluses, SEs, Classics) are too slow to handle 16->8 bit conversion in conjunction with the Sound Manager calls. Finally, the sound of computers equipped with the "Sound Manager" version 3.0 or better are automatically switched to the sound production routines that take advantage of this last, improved standard. (For more details on the three standards, see "Macintosh Sound" in Tutorial 3).
- ***External Synthesizers:*** You can also select an external sound synthesizer.

—The 411 synthesizer only works at the frequencies and bit widths supported by MacADIOS™ equipment. If necessary, convert your signal to a compatible frequency and bit width by means of the Complex Transformation in Transform>Complex Transformations.

—In the case of output via the "one-shot" MacADIOS II drivers, the entire signal is frequency- and bit width-adjusted prior to playback. This requires as much calculation memory as the signal itself. Be sure to reserve enough calculation memory through Edit>Startup Options.
—The Digidesign synthesizer drivers can handle any sound frequency and bit width without any noticeable load on calculation memory.

- ***Mono/Left channel/Right channel:*** Usually, the mono setting provides satisfactory reproduction. However, on some computers with some versions of the System (e.g., the Centris), an undesirable echo sound is heard with the mono setting. In that case, please choose the channel that is convenient.

Cautions

- **We do not guarantee the correct operation of routines for non-Apple synthesizers.**
- With some playback equipment, it is impossible to check if the external synthesizer is present or switched on and you must check yourself if output is functional. If it is not, Signalyze *may bomb*.
- With MacADIOS™ boards, you must specify the slot number. Do so through the Macro>Record Dialog.
- With some external devices, the specified sound level may not have any effect on the audio volume.
- In case of difficulties with the MacADIOS™ Model 411, turn off both computer and external system. Start the Model 411 first, then the computer, and try again.
- Audio output from Signalyze via the Digidesign boards requires that the DigiSystem INIT be in your System folder. As supplied, the INIT should run Ok. However, if the sound is reproduced too slowly, you should adjust the hardware setup. You have to start either the AudioMedia software (version 2.02 or later) or the Sound Designer II software (version 2.3 or later). In the menu "Hardware Setup", choose the following options. *For AudioMedia:* a. Peripheral: None b., Sample rate: **44100**, c. Sync Mode: Internal. *For Sound Accelerator:* a. Peripheral: (Whatever you have, AD-IN, etc.), b. Sample rate: **44100**.

Other Options:

— *Sound Level:* Controls the audio amplitude. Short cut: SHIFT-click on Sound Playback button (signal control panel).
— *Number of repetitions:* Single or multiple repetitions of a single stretch of sound.
— *Whole signal/Selected segment/Signal except for selection:*
—> With "whole signal", the entire signal is played, even if part of it is invisible. Short cut: OPTION-click on .

—> With “selected segment” only the blackened signal segment is played. If there is no blackening, the visible portion of the signal is played. Short cut: COMMAND-click on .

—> With “Signal except for selection” only the parts outside of the blackened signal segment are played. If there is no blackening, the visible portion of the signal is played. Short cut: OPTION-COMMAND-click on .

Signal>Play Signal ⌘ Y
Signal>Jouer signal ⌘ Y
Signal>Signalwiedergabe ⌘ Y

Menu equivalent for the Sound Replay Button on the Signal Control Panel ( ), to perform sound reproduction of the selected signal. This is handy if your cursor is in the neighborhood of the menu bar, or if you’d like to activate playback from the keyboard.

- On those devices that allow interruption (Apple, Digidesign), *interrupt sound reproduction* with COMMAND-period, COMMAND-click or click on Abort Button.

Signal>Delete Signal Display
Signal>Enlever affichage du signal
Signal>Signaldarstellung löschen

Deletes the *display* of the selected signal. The underlying signal remains in memory.

- To delete the display *as well as* the signal, use the Delete Signal menu command.
- To reestablish the display, use the “move signal” routine available through the Signal Selection button ( ).
- To delete the display with buttons, COMMAND-click on the Delete Signal button ( ).

Signal>Delete Signal ⌘D
Signal>Enlever signal ⌘D
Signal>Signal löschen ⌘D

Deletes the selected signal from memory. All displays of that signal and any spectrogram or spectrum based on the signal are erased.

- To delete the display without deleting the signal itself, use the Delete Signal Display menu command.
- Enable protective dialogs ("Save Signal? Yes/No") by setting the Signal>Delete Save Dialogs menu command (✓).
- To delete the signal with buttons, click on the Delete Signal button ( ).

Signal>Delete Signals on Card ⌘K
Signal>Enlever signaux de la carte ⌘K
Signal>Signale dieser Karte löschen ⌘K

Deletes all signals on presently visible card. **Caution**: Since the same signal can be shown on different cards, erasing it on the presently visible card may also cause it to be erased on cards that are not visible.

Signal>Delete All Signals ⌘L
Signal>Enlever tous les signaux ⌘L
Signal>Alle Signale löschen ⌘L

Deletes all stored signals and releases the entire signal memory.

Signal>Delete All But First ⌘1
Signal>Enlever tous sauf premier ⌘1
Signal>Alle ausser dem ersten löschen ⌘1

Deletes all signals on the current card except for the signal in the topmost signal track. **Caution**: Since the same signal can be shown on different cards, erasing it on the presently visible card may also cause it to be erased on cards that are not visible.

Signal>Enable Delete-Save Dialogs
Signal>Dialogues de précaution
Signal>Lösch/Speicher-Dialoge

This is a toggle to enable/disable the cautionary dialogs shown before deleting new and edited signals ("Save Signal? Yes/Cancel/No"). "No" and "Cancel" are equivalent in this dialog. Disabling the save dialog is particularly useful when operating with direct A/D input in an interactive (teaching, therapy) situation where saving signals is not generally desired.

Signal>New Start ⌘ ,
Signal>Etablir nouveau début ⌘ ,
Signal>Neuen Signalbeginn ⌘ ,

This command starts a multi-step procedure in which the user is asked to click at a new position where the signal is to start. This procedure is used to align a set of signals. Cancel the procedure any time by clicking on the Abort Button ( ). Set the display duration for the announcements through Edit>Preferences.

Signal>Original Start
Signal>Rétablir début original
Signal>Ursprünglichen Signalbeginn

This reestablishes the signal's original start position after a Signal>New Start command.

Signal>New MinMax ⌘ M
Signal>Etablir nouveau MinMax ⌘ M
Signal>Neuen MinMax ⌘ M

This command starts a multi-step procedure to resize the signal's vertical amplification within the display slot. The user is asked to first click on the new lower, then on the new upper limit to define the desired plot magnification. Cancel the procedure any time by clicking on the Abort Button ( ). Set the display duration for the announcements through Edit>Preferences.

This procedure is the *permanent* equivalent of a “vertical zoom”. For a *temporary* vertical zoom, please SHIFT-click on the -button.

Signal>Original MinMax
Signal>Rétablir MinMax original
Signal>Ursprünglichen MinMax

This reestablishes the signal’s original MinMax setting after a Signal>New MinMax command.

Signal>Synchronous Cursors ⌘=
Signal>Curseurs synchrones ⌘=
Signal>Synchrone Cursor ⌘=

When activated, this toggle menu command causes all line cursors in a multi-signal display to move in lock step. Activate again to “disengage” the line cursors’ time-synchronous action.

Signal>Show Whole Signal ⌘5
Signal>Montrer signal entier ⌘5
Signal>Gesamtes Signal zeigen ⌘5

This command causes the signal to be shown in such a way that it fills the display space. Key-click equivalent: SHIFT-COMMAND click on the Signal Plot Button ( ).

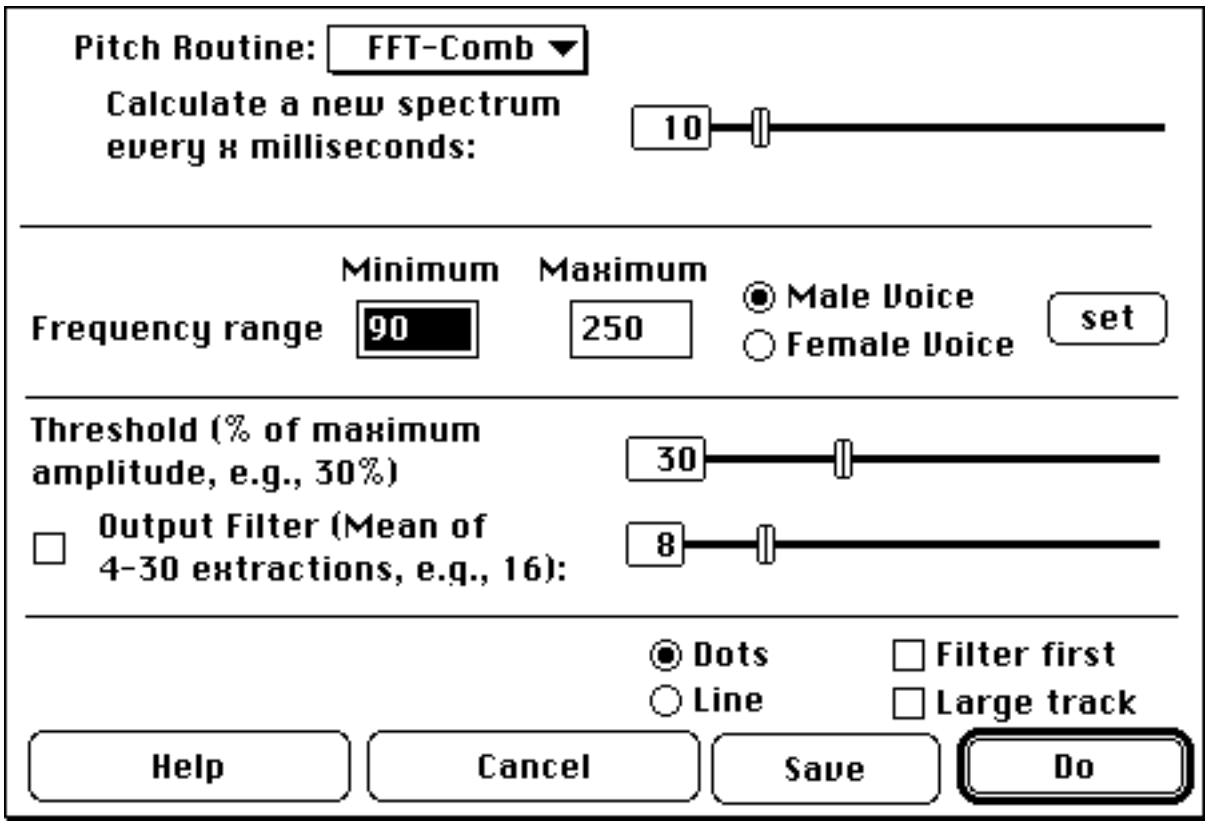
Signal>Show Selected Segment ⌘6
Signal>Montrer segment sélectionné ⌘6
Signal>Gewählten Signalteil zeigen ⌘6

This command causes the signal to be shown so that the selected signal segment fills the display space. Key-click equivalent: OPTION-COMMAND click on the Signal Plot Button ( ).

"Spectral" Menus

Spectral>Pitch Extraction Setup... ⌘7
Spectral>Options de fréquence fondamentale... ⌘7
Spektral>Optionen Grundfrequenzanalyse... ⌘7

CAUTION: All pitch extraction routines can make errors. Maintain a critical attitude towards pitch extraction results and cross-verify your results frequently.
Also, these pitch extraction techniques are only meant for speech. They may not work correctly for music.



How To
See Tutorial 5.

Specifications
- ***Choose a pitch extraction routine***

1. Temporal Structure Analysis (TSA). Fast, good for even, well-recorded voices. Provides an extraction for each pitch cycle.

But: Sensitive to large pitch changes. Frequency range must be well-defined and must not generally exceed much more than an octave.

2. FFT-Comb Routine. Default routine for most purposes. Reliable for many signals of various types. No need to specify upper frequency limit.

But: Not very good on low frequencies (< 90 Hz).

3. Autocorrelation. The routine of choice for "difficult" signals (noisy conditions, low pitch frequencies).

But: Slow, requires relatively close specification of frequency range.

- ***With FFT-Comb and Autocorrelation: Specify Interval of Spectrum Calculation:*** FFT-Comb and Autocorrelation depend on the calculation of a new spectrum every x milliseconds. Set the density of spectral calculations with the slider. Typical intervals are 5 ms or 10 ms. An interval of 5 ms is preferable if output filtering is chosen. An interval of 10 ms is frequently used in speech analysis.
- ***Frequency Range:***

TSA: You must estimate the frequency range quite closely, and it must not exceed much more one octave (maximum must be less than double the minimum).

FFT-Comb: Only the minimum frequency is of consequence to the extraction algorithm. The maximum frequency is used to limit the top of the display.

Autocorrelation: Both minimum and maximum frequencies are used by the extraction algorithm. However in contrast to TSA, the range need not be circumscribed as closely. Also, the range can be greater than one octave.

—*Male Voice, Female Voice, Set:* These features simplify switching between different typical frequency range settings. "Set" stores the specified frequency range for the selected type of voice.

- ***Threshold:*** Pitch extraction is only possible for voiced segments of speech, and "threshold" sets the differentiating criterion for the presence or absence of voicedness. The three routines use different criteria for setting the voiced/unvoiced threshold: TSA uses a signal slope criterion, FFT-Comb uses a combined intensity—low frequency criterion, and autocorrelation use an intensity criterion. Typically, values between 10% and 30% of maximum provide satisfactory voiced/unvoiced discrimination.
- ***Output Filter:*** Raw pitch curves obtained from any of the pitch extraction algorithms tend to contain extraneous and patently inaccurate data (so-called "outliers"). The output filter eliminates outliers and calculates a mean frequency on the basis of the retained data points.

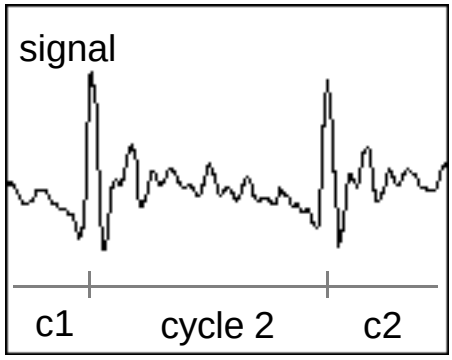
The routine works as follows: Given a series of x raw pitch extractions given in Hertz, the routine sorts the values and discards the upper and lower 25% of data points. On the basis of the remaining 50% of data points, a mean frequency is calculated. The pitch extraction curve

then connects the mean frequencies. The output filter routine can therefore be said to be based on a "median-mean" averaging algorithm.

Because of the nature of the algorithm, it is best to choose filter widths that are multiples of 4 (4, 8, 12, etc.). With the FFT-Comb or autocorrelation spectrum interval set at 5 ms, an output filter of 8 tends to give good results. With the TSA algorithm, greater filter widths (12, 16, 20) are possible. **CAUTION:** Temporal precision is seriously degraded in the FFT-Comb and autocorrelation routines if low interval frequency (e.g., 10 ms) is paired with great filter widths (>8).

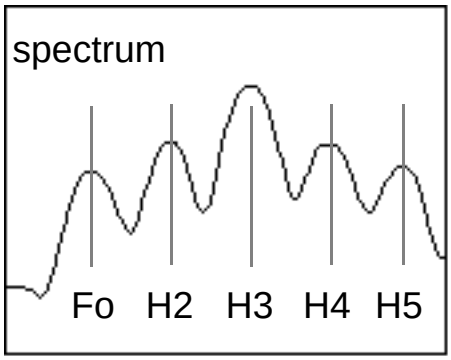
More on Pitch Extraction

The voice fundamental frequency (f_0, or *"pitch"*) is one of the most prominent correlates of the perceived intonation contour (other correlates: amplitude and vowel duration).



Conceptual basis of TSA:
Fo-related signal oscillations are large and regularly-spaced.

Conceptual basis of autocorrelation:
Segments in one cycle are very similar to (i.e., show strong positive correlation with) homologous segments in the next cycle(s).



Conceptual basis of FFT-comb:
The fundamental frequency and its harmonic frequencies are found at regular frequency intervals and thus "reinforce" each other.

There are a number of ways to extract pitch from a speech signal by digital algorithms (for an extensive review of the question, see Hess, 1983). Among the most common are, on the one hand, techniques based on Fast Fourier Transforms and autocorrelation. On the other hand, there are time-domain pitch detectors which include temporal structure analysis. The conceptual bases of the three types of extraction are illustrated below.

There exists a rough tradeoff between time requirements and the robustness and reliability of the procedure. For example, autocorrelation-based techniques are relatively robust, but can be fairly time-consuming

on microcomputers without specialized hardware. On the other hand, techniques based on temporal structure analysis ("peak-to-peak period detectors") tend to be less robust, but much faster under most circumstances.

Temporal Structure Analysis (TSA)
vs. FFT-comb Analysis
vs. Autocorrelation Analysis

• ***TSA*** attempts to identify strong falls and rises characterizing the onset and termination of large oscillations found at fairly regular intervals in voiced sections of acoustic speech signals. The user-set frequency range specifies limits for the window within which the search for strong fall-rise patterns is conducted.[1]

Once the set of strongest fall-rise patterns has been identified, time intervals between the *end points* of identified patterns are converted into raw Fo frequency values.

—Advantages: TSA follows voice modulations quite closely, since an extraction is attempted for every voice period. The routine is insensitive to divergences between fundamental frequency and harmonic frequencies, since it disregards the harmonic information.

—Disadvantages: Unless the frequency range is tightly specified, obtained values are likely erroneous. Also, harsh or irregular voices play havoc with the identification of fall-rise patterns.

• An ***FFT-Comb*** analysis performs successive 512-point FFTs, and frequency information from both Fo and its harmonic frequencies are combined to determine the Fo contour[2].

The combining algorithm proceeds as follows: Parabolic cones are calculated around each of the fundamental and harmonic frequency responses in the FFT output and their forms are cumulated. Fo frequency is given by the peak of the largest cumulative cone. Signals with sampling frequencies above 20 kHz are subsampled prior to the application of the FFT.

—Advantages: No need to circumscribe the frequency range very closely. Rather reliable in normal speech. Good distinction of voiced and unvoiced periods.

—Disadvantages: Relatively slow. Also, the FFT window imposes a filter that smooths out rapid voice modulations. It is sensitive to divergences between fundamental frequency and harmonic frequencies that occur in some voices. May be unsuitable for analysis of voice disorders because of

---

1 The current Signalyze implementation of this routine is based on on-going research. This is not necessarily the final version of the algorithm.

2 The original form of the present FFT-comb routine was proposed by Philippe Martin of the University of Toronto.

the temporal imprecision and the sensitivity to divergence between Fo and harmonic frequencies.

• An ***autocorrelation analysis*** extracts a set of autocorrelation-based spectra, and Fo is determined from the strongest peak in the user-defined frequency window.

In this algorithm, the amplitude values of a given stretch of signal are correlated with those of succeeding signal segments. Two segments lying in comparable portions of the fundamental frequency cycle provide high, positive correlations, while two segments from different segments of the cycle show low or negative correlations.

The Signalyze implementation of the algorithm seeks only the first, strongly positive correlation (i.e., the correlation between adjoining cycles). Other implementations verify this result by calculating correlations over two or more cycles. However in the context of an interactive program working on relatively small machines, such an approach would not be practicable.

Signals with higher sampling frequencies are subsampled prior to the application of autocorrelation analysis by a factor determined by a relatively complex formula relating sampling rate to the user-specified low frequency bounds and to the width of the correlational analysis window.

—Advantages: Rather reliable over a wide range of conditions, including noisy background. Responds only to fundamental frequency and ignores harmonic frequencies. Good distinction of voiced and unvoiced periods.

—Disadvantages: Rather slow. Also, the autocorrelation window imposes a filter that smooths out rapid voice modulations. May thus be unsuitable for analysis of voice disorders.

Verification of Pitch Extraction Results

Verify the correctness of an extracted pitch contour as follows:

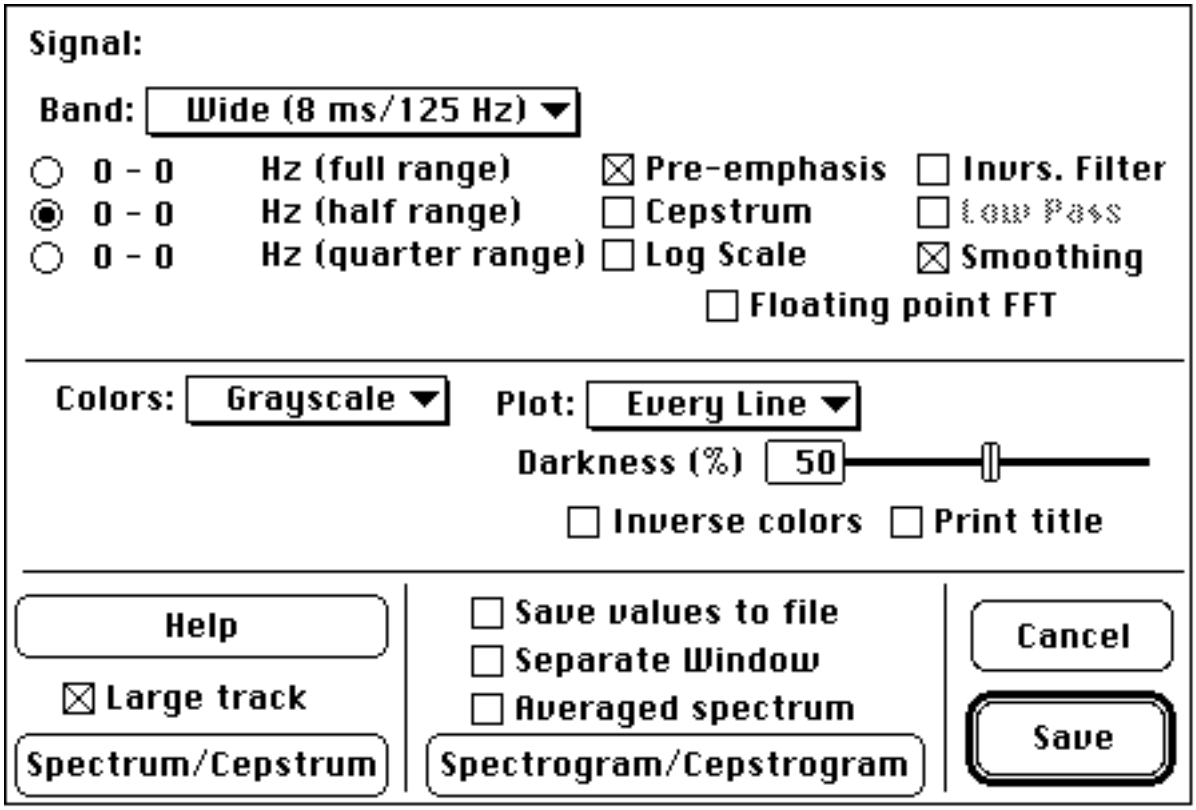
- Zoom a voiced portion of the signal with fairly stable Fo contour.
- Blacken a peak-to-peak section.
- Read off number in the the upper right part of the signal window, marked with "f". This is the frequency equivalent of the duration of the blackened section of the signal.
- The frequency of the Fo contour (visible to the left of the signal) should match the frequency equivalent within a few Hertz.

Spectral>Extract Pitch... ⌘7
Spectral>Extraction de fréquence fondamentale... ⌘7
Spektral>Grundfrequenz... ⌘7

Initiates pitch extraction according to settings specified in Spectral>Pitch Extraction Setup.

Spectral>Spectral Analysis Setup... ⌘8
Spectral>Options d’analyse spectrale... ⌘8
Spectral>Optionen Spektralanalyse... ⌘8

IMPORTANT NOTICE: Though you may be able to get frequency read-outs by clicking in spectrograms, you should never use such values in a research project. The only reliable read-outs are those provided by clicking in *spectra* or *cepstra*.



The Spectral Analysis Setup specifies values for spectrograms, cepstrograms, spectra and cepstra.

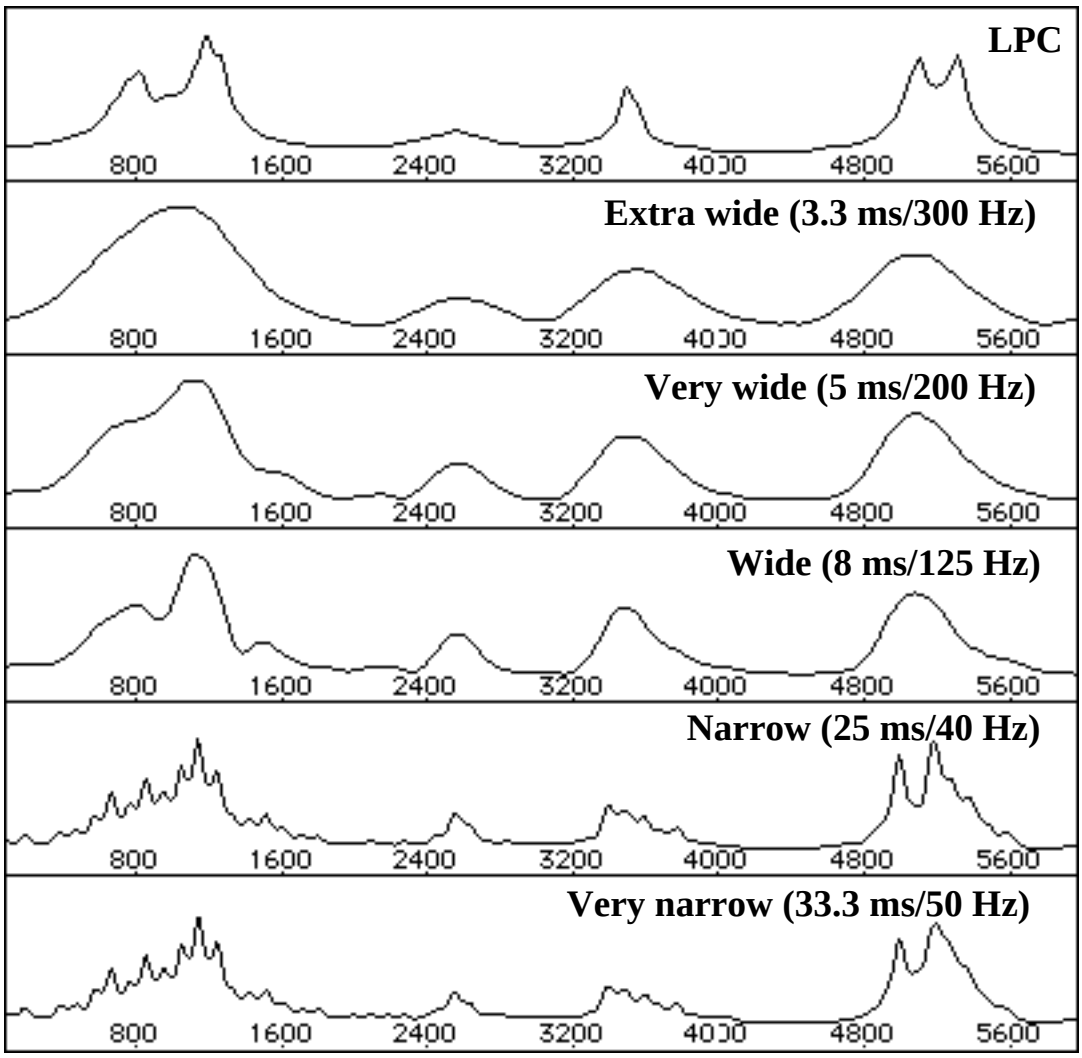
CAUTION: Even though the settings are the same in spectra/cepstra and in spectrograms/cepstrograms, the algorithms used are somewhat different. The spectral/cepstral algorithm uses a 1024-point

FFT, while the spectrogram/cepstrogram algorithm uses 256-point and 512-point FFTs to increase display speed.

How To

See tutorial 5.

Top Panel Specifications: Algorithm Settings



Spectral analysis on an open /a/, male voice, 16 kHz sampling rate. Notice that only wide-band analysis (band-width of 125 Hz) and LPC distinguish the first two formants correctly (at 800 Hz and 1200 Hz). This is one reason why Signalyze sets the default for wide-band analysis at 125 Hz, in contrast to some other systems that set it at 300 Hz. However, Signalyze offers a wide range of band widths for analysis.

- **Band:** Specifies the width of the frequency band over which signal energy is measured. See below for more detailed explanations of how the frequency bands are determined.

Very slim band: 100 ms of signal material is analyzed, giving a bandwidth of 10 Hz. Appropriate for the analysis of music and of prolonged animal sounds.

Slim band: 50 ms of signal material is analyzed, giving a bandwidth of 20 Hz. Appropriate for the analysis of music and of prolonged animal sounds, sometimes useful for Fo analysis of men's voices.

Very narrow band: 33.3 ms of signal material is analyzed, giving a bandwidth of 30 Hz. Appropriate for the analysis of Fo and harmonic frequencies of men's voices.

Narrow band: 25 ms of signal material is analyzed, giving a bandwidth of 40 Hz. Appropriate for the analysis of Fo and harmonic frequencies of women's and children's voices.

Wide band: 8 ms of signal material is analyzed, giving a bandwidth of 125 Hz. Appropriate for the analysis of the formants of men's voices.

Very wide band: 5 ms of signal material is analyzed, giving a bandwidth of 200 Hz. Appropriate for the analysis of the formants of women's and children's voices.

Extra wide band: 3.33 ms of signal material is analyzed, giving a bandwidth of 300 Hz. Sometimes appropriate for the analysis of the formants of women's and children's voices.

LPC: An LPC-type spectral analysis is performed. Appropriate for the analysis of any type of voice.

Cone kernel: A cone kernel-type spectral analysis is performed. Appropriate for the analysis of any type of voice.

- **Order (only with LPC):** Specifies the number of autocorrelations forming the basis of LPC frequency analysis. Here are the standard recommendations for LPC analysis:

- Respect *anti-aliasing constraints* when sampling the signal (for most digitizers, this is automatic—see filtering in Tutorial 7, as well as the section "More on Spectral Analysis", below).
- Use *sampling rates* as low as possible, i.e., just above twice the highest expected formant frequency.
- *Order (coefficient) number:* Take 2 orders per formant-containing kHz, plus 2 or 3. The 2 or 3 extra orders model the laryngeal pulse. First approximations generally work out as follows: for males, take 2 orders, for females 1 order per kHz below the Nyquist frequency, plus 2 or 3.
  Example: Suppose you want to see four formants in a male voice. The highest expected formant frequency is 5 kHz. Sample at 13 kHz, with an anti-alias setting of 5.2 kHz (40% of 13 kHz). There are 6 kHz below the Nyquist frequency of 6.5 kHz, so calculate 2*6=12, add 2 or 3 to get 14 or 15.
  If you use too many coefficients: You get spurious peaks around harmonic frequencies.

If you use too few coefficients: Several peaks coalesce into broad "mountain ranges".

- Since the number of coefficients (orders) is relatively small, the *chances of error* in the model are relatively high. For this reason, LPC extractions are quite often "strange" (no peaks, weird-looking peaks, or peaks in patently wrong places). Nudge the cursor over just a bit and try again. Chances are you'll do much better.

Experiment with different orders at the sampling frequency you typically use. *Order number **must** be fitted to the sampling frequency: Compare your results with those of wide-band and narrow-band FFTs.* Initially, you may want to reduce your sampling frequency to 10 kHz by subsampling (Transformation> Complex Transformations) and use order number 14. This generally gives satisfactory extractions.

CAUTION: USE LPC WITH CARE. LPC formant measures often diverge to a substantial extent from narrow-band or wide-band formant measures. Some laboratories have ceased to use LPCs after years of using it. Still, LPCs can be useful to distinguish two closely situated, adjoining formants (as F1 and F2 in the vowel /a/), especially in cases where wide-band analysis cannot show the distinction. Also, LPC analysis is relatively insensitive to differences between male and female voices. The rule is: Above all, fit your order number to your sampling frequency.

- **Form (only with cone kernel):** Specifies the type of autocorrelation that forms the basis of the cone kernel analysis.

— Basic cone: Cone kernel autocorrelation built up of ever-smaller, superimposing portions of a given signal segment.

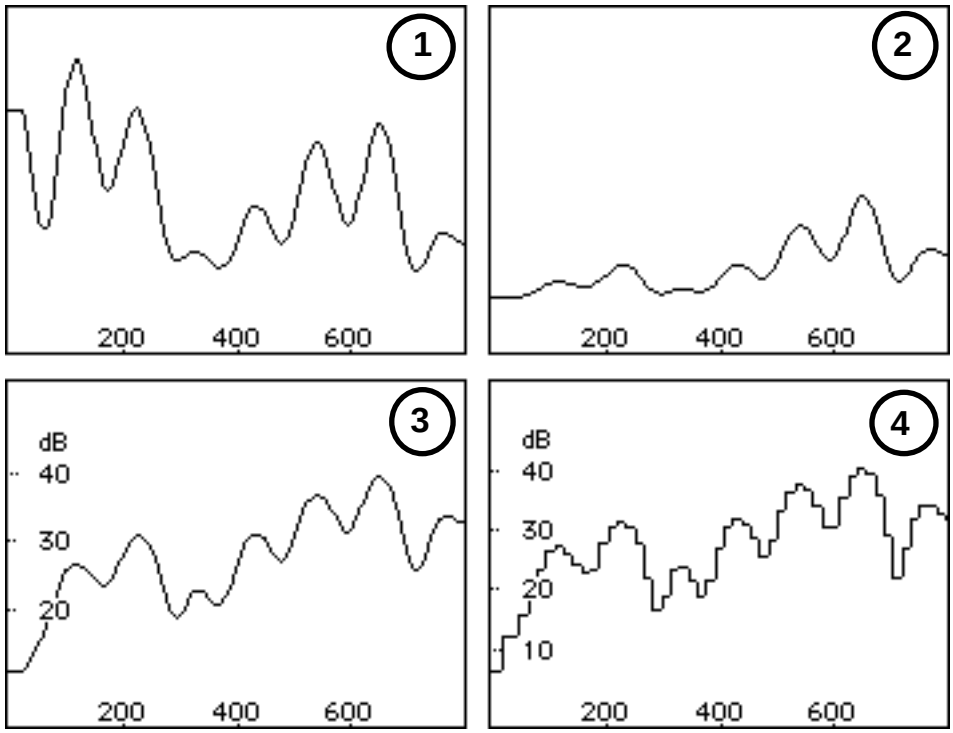
— Double diamond: Cone kernel autocorrelation built up of two passages of the same, short signal segment, whereby the analyzed portion first increases, and then decreases in size during each of the two passages.

- **Range:** Specifies which part of the spectral analysis result is shown. The range specification does not influence the FFT analysis, it merely controls the zoom of the display.

—Full range: The entire spectrum is shown, extending from 0 Hz to 1/2 sampling frequency (the so-called "Nyquist frequency"). Useful for wide-band spectra and cepstra.

—Half range: Half of the spectrum is shown, extending from 0 Hz to 1/4 sampling frequency.

—Quarter range: A quarter of the spectrum is shown, extending from 0 Hz to 1/8 sampling frequency. Useful for narrow-band spectra.



Emphasis, log, smoothing: The same narrow-band spectrum, taken in an example of the vowel /∂/, shown with different display options: (1) no pre-emphasis, no log, smoothing, (2) pre-emphasis, no log, smoothing, (3) pre-emphasis, log, smoothing, (4) pre-emphasis, log, no smoothing.

- **Pre-emphasis:** The lower frequencies tend to show greater amplitudes than the higher frequencies, giving the descending "organ pipe effect" typically seen in raw spectra and known as "spectral tilt". The pre-emphasis option counteracts this descending gradient by "boosting" higher frequencies more than lower frequencies. In Signalyze, this is achieved by using differences between adjoining samples instead of raw signal values as input to the FFT. This provides an amplitude boost of roughly 6 dB per octave. Default setting: **On**. Should be **off** with spectra showing strong positive tilt.
- **Inverse Filter:** Applies an autocorrelation inverse filter prior to spectral analysis by FFT. This often results in smoothed spectral output and emphasized formant peaks, particularly useful for examining the first two formants. This option should only be on when formant analysis is performed.

This filter is different from the LPC in that it is based on the output of a large set of autocorrelations, while the LPC abstracts the autocorrelation information further into a small number of points prior to performing the FFT. The "small number" is that given by the LPC's order setting.

Usual setting for formant frequency analysis using inverse filtering for generating a spectrum:

narrow-band spectrum	pre-emphasis on
inverse filter on	smoothing on

Sometimes it may be better to switch to wide-band spectrum in order to reduce the number of peaks. Default setting: **Off**.

CAUTION: The inverse filter may sometimes cause formant frequency peaks to shift or to disappear.

- **Log scale:** Calculates the log10 of spectral values after spectral analysis. This has the effect of attenuating differences between high and low amplitude spectral peaks and of displaying spectral amplitude according to the decibel scale.

This option should generally be off for spectrograms (possible exception: narrow-band spectrograms, where the log scale may sometimes enhance visibility of Fo and harmonic frequency bands). It may be on for spectra, especially if a y-scale dB read-out is desired (see Scale Setup)[1].

- **Cepstrum:** When selected, a cepstrum is drawn instead of a spectrum, or a cepstrogram is drawn instead of a spectrogram. See below for more details about cepstra.

Cepstra are generally used with narrow band spectral analysis to identify Fo. The frequency range should initially be set to "full range" or "half range", otherwise the cepstral peak for Fo may be out of display range.

- **Low Pass:** With a cepstrum, the low pass setting enhances the visibility of the fundamental frequency peak by suppressing high-frequency-related peaks in the cepstrum. Default setting: **On**.
- **Smoothing:** Applies a very rapid 3-sample triangular smooth to spectral output. With smoothing off, the spectrum appears as it emerges from the FFT. Default setting: **On**. *PLEASE NOTE:* Decibel or numeric values obtained from the spectrum by clicking or via the statistics command are always based on the *unsmoothed* spectrum.
- **Floating-point:** When selected, the program uses the floating-point version of the FFT. Should be selected by default on all faster machines or when high precision is required.

Mid Panel Specifications: Spectrogram Settings

- **Resolution:**

"Black & white" — two-color, dithered grays (old-style Mac)

---

[1] "Real" speech scientists often say that log should be on all the time because log output conforms to the decibel scale. I personally prefer to play it by ear. With wide-band spectral analysis, log seldom seems useful. With narrow-band analysis, it can sometimes help in visualizing Fo. But then, turning off pre-emphasis often achieves the same end—and distorts the data less.

“Grayscale”, “Colors” — grayscale or color (new-style Mac)
Notes:
—If you want scales and/or gridlines, specify your selections in Scales Setup *before* drawing a spectrogram.
—When choosing “Black & white”, you can choose dithering pattern with the 2-5-tone Shading popup.
—If desired, adjust colors or gray levels with Edit>Spectrogram Palette.
—When choosing “low resolution”, you can choose the dithering pattern with the 2-5-tone Shading popup.

- **Spacing:**
  “Plot every other line” — fast overview.
  “Plot every line” — publication quality.
  “Repeat every other line” — faster.
- **Darkness (%):** The usual setting is 10%-50%. Use higher settings for weak signals or signal segments.
- **Inverse colors:** Usually, high amplitudes are drawn as dark colors. With “Inverse colors” selected, high amplitudes are drawn as light colors and low amplitudes are drawn as dark colors. (Remember: you can adjust the spectrogram palette with Edit>Spectrogram Palette).
- **Print title:** Uses the signal name as a title on spectrograms. Specify the title’s position in Spectral>Plot Setup. Change the title by SHIFT-clicking on the signal name in the Signal Control Panel.

Bottom Panel Specifications: Output Settings

- **Left:** Spectrum/Cepstrum on normal-sized or large signal track.
- **Middle:**

—*Save values to file:* If selected, the program saves the output of the spectrogram to a (very large) ASCII file.
— *Separate window:* Instead of drawing the spectrogram on the Signal Window, create a separate window and draw the spectrogram twice the size.
— *Averaged Spectrum:* Selecting this option calculates a long-term averaged spectrum (short: “averaged spectrum”) in conjunction with the spectrogram. After a spectrogram is drawn, the corresponding averaged spectrum is displayed. It represents an average of the same FFT values that are used to draw the spectrogram, i.e., it is based on a 256-point or 512-point FFT (in contrast to simple spectra which are based on 1024-point FFTs). Averaged spectra can be drawn up indifferently from spectrograms, cepstrograms or LPC-grams.

- **Right:** Cancel or save the settings.

Printing out of Spectrograms

See Tutorial 8.

Darkness Adjustment and Stopping Spectrogram

- **In the Signal Window:** Set darkness ahead of time through SHIFT-click on darkness adjustment popup (below the Abort button) or in the Spectral>Spectral Analysis Setup dialog.
- **Separate Window:** Set darkness also with the slider below the spectrogram during spectrogram generation. The routine "comes around once more" to update the spectrogram fully after an adjustment.
- **Abort in the usual three ways:**
  —COMMAND-click
  —COMMAND-period (.)
  —Click on Abort button

More on Spectral Analysis

Preliminaries

A number of considerations must be taken into account when performing Fast Fourier Transform-based spectral analysis.

Anti-aliasing: To prevent the introduction of erroneous high frequency components, the speech signal must be low band-passed at half the sampling frequency with a so-called "anti-aliasing filter" *before* spectral analysis is performed. For example, a signal recorded at 10 kHz must be filtered in such a way that frequencies above 5 kHz are either strongly attenuated or eliminated. Set the filter to somewhat *below* the half-frequency because of filter roll-off. 40% of sampling frequency is a conservative setting. For more details about filtering, and anti-aliasing filtering in particular, see Tutorial 7.

Preferably, use 12-to-16-bit signals. This precision provides sufficient dynamic range for high-quality spectrograms. Somewhat less informative and grainier spectrograms are generated from signals that are recorded at 8-bit precision. Imprecisions typically affect the higher frequency components much more than lower frequency components.

The Frequency Precision—Frequency Range Tradeoff. At a given type of analysis, spectral analysis of signals with low sampling rates gives more precise results than the analysis of high sampling rates. For example, the rays (responses) of a 1024-pt FFT are spaced at ±10 Hz with a 10 kHz signal, while they are spaced at ±20 Hz with a 20 kHz signal. You

may wish to subsample your signals with the Transform>Complex Transformation prior to spectral analysis.

Spectral Frequency and Amplitude (Intensity)

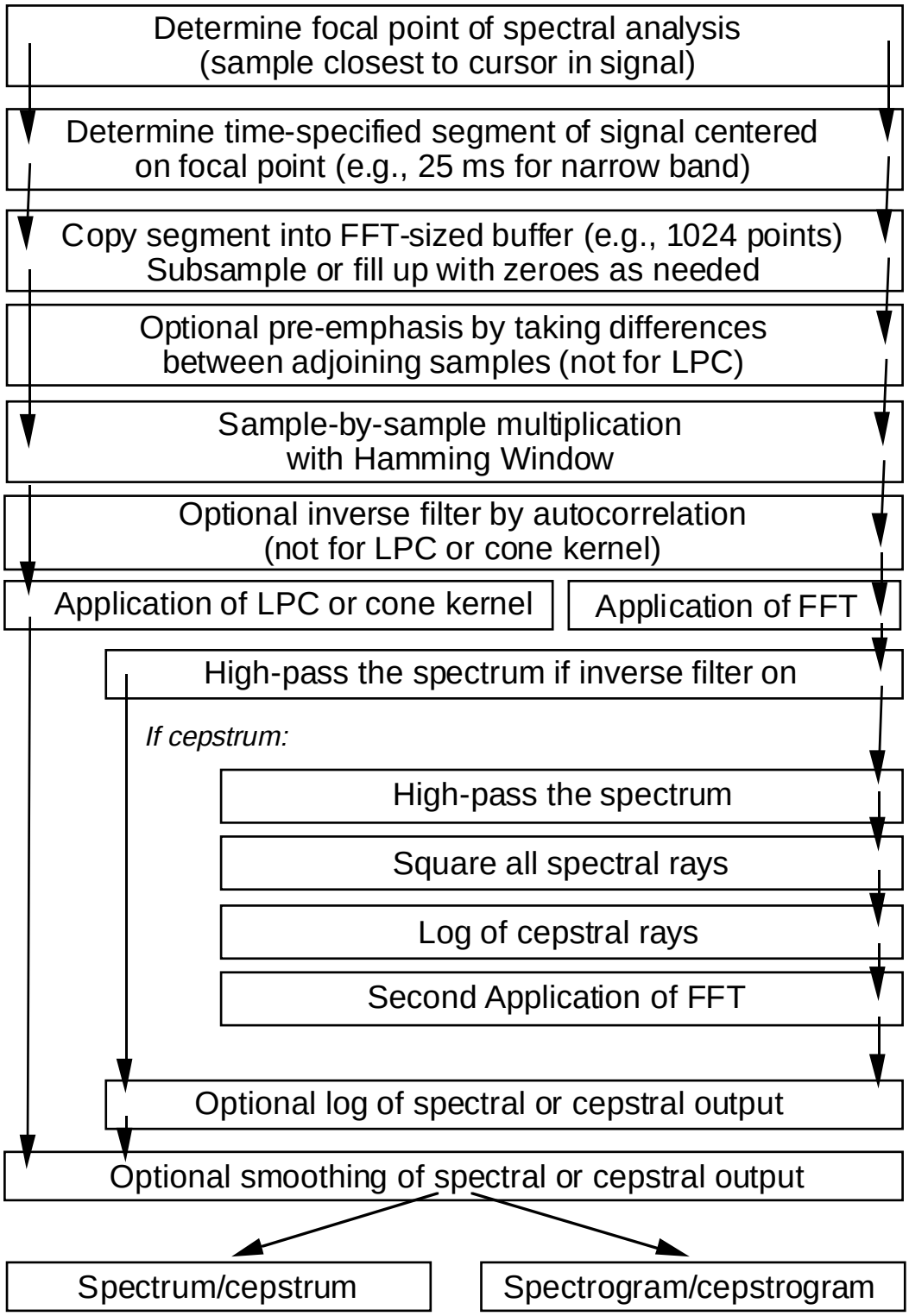
- *Where to read frequency and amplitude:* For spectra, Signalyze shows frequency values at the cursor in the upper numeric indicator, and relative dB amplitude values in the lower numeric indicator of the Signal Control Panel. The values can be transferred to the Edit Window by clicking on the indicator, or by clicking in the spectrum while pressing SHIFT-COMMAND (shows frequency and time in signal where the spectrum was taken), or while pressing SHIFT-OPTION-COMMAND (shows relative dB values). The numeric format can be set with Signal>Numeric Setup.
- *Frequency and amplitude differences:* Relative frequency and amplitude differences in a spectrum can also be obtained through Edit>Show Statistics. Check the appropriate box(es) in Edit>Show Statistics Options.
- *Calculate decibels offline:* Users who wish to convert spectral output into decibel information outside Signalyze can proceed as follows:

—Blacken the spectral segment between two peaks.
—In Edit>Show Statistics Options, select "left endpoint value" and "right endpoint value". Exit with "Do".
—Raw spectral output values are shown in Edit Window. Press RETURN to transfer values to Text Window.
—Save the values to a text file in the usual manner.
—In a separate spreadsheet program, calculate the dB difference between the two endpoint values with the following formula:

x = amplitude of left raw FFT end point value
y = amplitude of right raw FFT end point value
$dB_X = 20 * \log10(x)$
$dB_Y = 20 * \log10(y)$
dB difference = $dB_X - dB_Y$

PLEASE NOTE: Selecting "smoothing" in Signal>Spectral Analysis Setup does not affect the numeric values furnished in manual scoring or through the statistics commands. Numeric values always reflect the raw FFT output, whether translated into relative dB or reported as raw FFT output values.

A Peek under the Hood: How Signalyze Spectral Analysis is Performed



General flow chart of spectral analysis

Overall sequence of operations: Calculation of a spectrum or cepstrum follows the sequence illustrated in the flow chart above. Essentially, there are three paths:

- Spectral analysis via FFT.
- Spectral analysis via LPC or cone kernel.
- Cepstral analysis via double application of FFT.

The various user options are discussed above. In this section, operations are discussed that are are hidden from view, yet are important to the understanding of spectral analysis.

Location and Width of Analysis window: Initially, the sample lying closest to the cursor is located, and an analysis window centered on this sample is calculated. The width of the analysis window is given by the user's choice of band width. Spectral analysis at narrow band widths identify both overall features (formants and bands of frication energy) and finer features (fundamental frequency and its harmonics). At wider band widths, the analysis "glosses over" the finer features and the spectral response corresponds primarily to overall formant and frication energy features.

Narrow band widths require relatively long analysis windows (e.g., a 40 Hz band requires 25 ms of signal material), while wide band widths involve the analysis of short analysis windows (e.g., a 125 Hz band requires 8 ms of signal material). The band widths selected for Signalyze correspond in part to traditional band widths and in part to those found particularly useful in the authors' experience.

Hamming Window: If the signal segment were analyzed exactly as it is taken from the signal, the spectral analysis routine would identify an exceedingly strong high frequency component corresponding to the on- and offset of the segment (the so-called "edge effect"). For this reason, the selected signal segment has to be weighted so as to be weaker at the edges than in the middle.

The standard approach is to multiply the window's values with the values of a so-called "hamming" curve. This imposes a bell-shaped weighting on the window, so that signal values at the two edges of the analysis window are attenuated to the signal's average amplitude, while the middle is left at its original amplitude. Signalyze automatically applies a hamming window before performing a spectral analysis on a given analysis window.

Why Control Analysis Window Width and not FFT Width?

In contrast to some other programs (e.g., SoundEdit), Signalyze lets you control analysis window width (in ms), and not FFT width. Why?

Spectral analysis by digital means offer a tradeoff between temporal and spectral precision:
Narrow FFT width: Great temporal precision, low spectral precision.
Wide FFT width: Low temporal precision, high spectral precision.
Moreover, precision interacts with sampling frequency. E.g., a 128-point FFT on a 22 kHz signal provides different temporal and spectral precision than a 128-point FFT on a 16 kHz signal.

In speech analysis, *comparable spectral and temporal precision* is required, no matter what the sampling frequency. A "125 Hz wide-band analysis" or a "30 Hz narrow-band analysis" should provide similar results over a wide range of sampling rates.

This requirement poses a problem for fixed-length FFTs, since they operate only on signal segments that are powers of 2 (e.g., 128, 256, 512, etc. samples long). If two signal segments of the same length, but from signals with different sampling rates were taken into the same FFT, the frequency results would not be directly comparable. Signalyze thus does not directly control FFT length, but rather, controls how much data is taken into the FFT. If two sets of samples representing a time window of the same duration are taken into a given FFT, comparability of results across sampling rates is much improved.

Some adjustments are necessary with this approach. If the time window is smaller than FFT width, "empty positions" in the FFT have to be filled with zeroes. If it is larger than FFT width, subsampling is employed to "shoehorn" the data into the FFT. Perhaps surprisingly, the result is quite satisfactory, and comparability over a large range of sampling rates (roughly 5 - 80 kHz) is assured. Although the extra zeroes do induce some extra noise due to the FFT's reduced amplitude output, smoothing largely compensates these imprecisions.

As another advantage, this manner of proceeding lets Signalyze easily switch between 256-point, 512-point and 1024-point FFTs to accelerate spectral analysis. Wide-band spectro/cepstrograms can use the fast 256-point FFT, narrow-band spectro/cepstrograms can use a moderately fast 512-point FFT and for high-precision spectra or cepstra, a 1024-point FFT is employed. The 1024-point FFT takes a bit longer, but then that's ok, since only a single FFT is generated.

The Cepstrum: By selecting the cepstrum option, the user can obtain an analysis that has considerable usefulness in speech analysis. The term "**cep**strum" was created by reversing the first four letters in "**spec**trum", to indicate the fact that this type of analysis is the inverse of spectral analysis. The basic concepts of the cepstrum are explained in Tutorial 5.

For good results when working with higher frequency responses, cepstral analysis is performed, not on the original FFT output, but on the log of the squared values of the FFT output. In order to attain acceptable speeds with a 16-bit integer version of the FFT on a microcomputer, the output of the first FFT is scaled down to a level where squaring is possible without integer overflow. The scaling varies from one FFT to the next, with greater scaling applying to the stronger amplitude segments of the signal. For this reason, the cepstrum appears somewhat lighter in the strong amplitude sections and darker in weak amplitude sections.

Spectrogram generation: Signalyze generates a new FFT for each pixel line, or if selected by the user, for each second pixel line. Intensity is calculated for the various frequency components and is scaled to the number of gray or color scales the user has selected, or that the video board is capable of. FFT results are displayed in the appropriate grayness or color levels on the vertical pixel line.

For pixels lying close to the beginning or the end of the signal, the analysis window might extend beyond the signal's limits; in those cases, the spectrogram routine begins drawing lines only half an analysis window width into the signal, or stops drawing half an analysis window width before the end of the signal. This leaves a margin of white space on either side of the signal, an effect that is particularly visible with great zoom amplification.

Window Overlap: For the next vertical pixel line, the analysis window is displaced to the right by the appropriate number of samples, and the whole process is repeated until either the end of the signal or the right edge of the display window is encountered. Depending on sampling rate, zoom level and number of FFTs per pixel line, the various analysis windows may or may not overlap.

Spectra ("Spectral Sections", or "Slice Plots"): In order to reduce analysis imprecisions in measuring the fundamental frequency, you can measure a strong harmonic frequency (e.g., the 5th harmonic) and divide the frequency by the appropriate number (e.g., 5).

Spectral>Spectrogram-Cepstrogram...

Spectral>Spectrogramme-Cepstrogramme...
Spektral>Spektrogramm-Cepstrogramm...

Initiates spectrogram or cepstrogram generation in accordance with specifications in the Spectral>Spectral Analysis Setup dialog. Shortcut: Click on button.

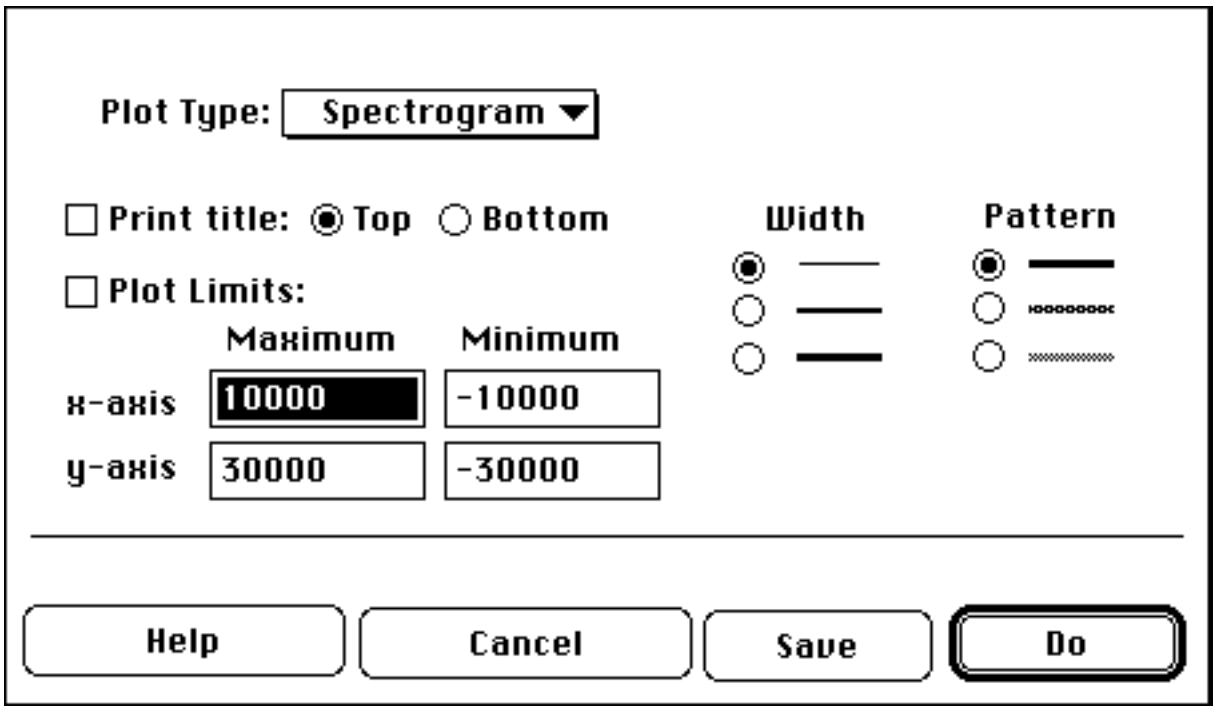
Spectral>Spectrum/Cepstrum...

Spectral>Spectre/Cepstre...
Spektral>Spektrum/Cepstrum...

Initiates spectrum or cepstrum generation in accordance with specifications in the Spectral>Spectral Analysis Setup dialog. Shortcut: Click on button.

Spectral>Plot Setup...

Spectral>Options de graphique...
Spektral>Optionen Graphik...



General Information

Permits the display of some special, large-sized plots of detailed signal segments, as well as some specialized plots.

- These plots can be resized (1) by resizing the window, and (2) by clicking on the upper left resize button ( ).
- The time indicator (usually ms) can be changed in Edit>Numeric Formats.
- The time and X-Y plots show as many points as there are pixels in the window. When there are more samples in the selected portion of the signal than pixels, points are selected by sampling the signal segment at equal intervals.
- You can save this type of plot as an image file and you can print it out.
- *Please note:* These plots operate in floating point and are therefore quite slow.
- **Plot type:** Choose type of plot.
- **Print title:** Prints the signal's name either over or below the plot. Change the title by SHIFT-clicking on the signal name in the Signal Control Panel.
- **Width and Pattern:** Adjust the width and the grayness pattern for drawing the line.
- **Maximum and Minimum:** If you enable the check box "Plot Limits", you can set the maximum and minimum values yourself. If you leave the box unchecked, the maximum and minimum values will be adjusted automatically. Manual setting of these values permits drawing plots that are always dimensioned in the same way, and that are thus comparable.

Time Plot

This makes a blow-up of a selected signal segment. This is often useful for teaching purposes.

- Set "Time Plot" in Spectral>Plot Setup.
- Select (blacken) a signal segment by dragging in the signal.
- Start the plot with Spectral>Make Plot.

X-Y Plot

This plots values in a first signal on the x axis against values in a second signal on the y axis. This plot is particularly useful in establishing differences between closely related signals, such as the two channels in a stereo recording.

- Set "X-Y Plot" in Spectral>Plot Setup.
- Select the signal for the x axis with the First Signal Button ( ).
- Select the signal for the y axis Second Signal Button ( ).
- Blacken a signal segment in either the first or the second signal.
- Start the plot with Spectral>Make Plot.

- If the selected segments of the two signals are of unequal length, the intersect of the two selections is plotted.
- For maximally useful plots, make sure that both signals are at the same sampling rate.

Phase Plot

The phase plots momentary amplitude on the y axis against momentary velocity (differences between adjoining amplitude values) on the x axis. Negative velocities are to the left of the midline, positive velocities to the right.

- Set “Phase Plot” in Spectral>Plot Setup.
- Select (blacken) a signal segment by dragging in the signal.
- Start the plot with Spectral>Make Plot.
- The graph is scaled to fill the window.
- Phase plots show every sample of the selected signal portion.
- Contrary to established practice, Signalyze phase plots invert the x and y axis. This has the advantage of keeping the y-axis comparable to that of the underlying signal.

Spectrogram

For ease of operation, the spectrogram request is repeated here. Spectrogram setup occurs through Spectral>Spectral Analysis Setup.

Spectral>Make Plot

Spectral>Afficher graphique
Spektral>Graphik

Initiates the generation of a plot in accordance with specifications in Spectral>Plot Setup.

“Window” Menus

Window>Hide Signal Window

Fenêtre>Cacher fenêtre de signal
Fenster>Signalfenster verbergen

This toggle causes the Signal Window to disappear temporarily. Activating the command a second time causes the window to be shown

---

again. This is useful for working on other windows or programs temporarily.

Window>Hide Text Window

Fenêtre>Cacher fenêtre de texte
Fenster>Textfenster verbergen

This toggle causes the Text Window to disappear temporarily. Activating the command a second time causes the window to be shown again. This is useful for working on other windows or programs temporarily.

Window>Hide All Windows

Fenêtre>Cacher toutes les fenêtres
Fenster>Alle Fenster verbergen

This toggle causes all Signalyze windows to disappear temporarily. Activating the command a second time causes the windows to be shown again. This is useful in multi-application operation, to permit viewing windows lying below the Signalyze windows.

Window>Hide on Context Switch

Fenêtre>Cacher en multitâche
Fenster>Bei Programmwechsel verbergen

This toggle causes all Signalyze windows to disappear as the user clicks on the window of a different program when operating in a multi-application environment. They reappear when switching back to Signalyze. Since the Signal Window frequently covers the whole screen, it’s often good to have it out of the way.

Window>Optional Menus...

Fenêtre>Menus optionnels...
Fenster>Fakultative Menüs...

Gives the user control over the visibility of the following menus:

- Transform Menu
- Macro Menu

- Font & Size Menus

This is often useful when working on 9-inch screens of the Classic Mac, Mac SE and MacPlus series. On program startup, the Transform and Macro Menus are always reestablished, so as not to confuse the user.

Window>Signal Window ⌘9
Fenêtre>Fenêtre de signal ⌘9
Fenster>Signalfenster ⌘9

This command brings the Signal Window to the front, if it is not already in front.

Window>Text Window ⌘0 (zero)
Fenêtre>Fenêtre de texte ⌘0 (zéro)
Fenster>Textfenster ⌘0 (Null)

This command brings the Text Window to the front, if it is not already in front.

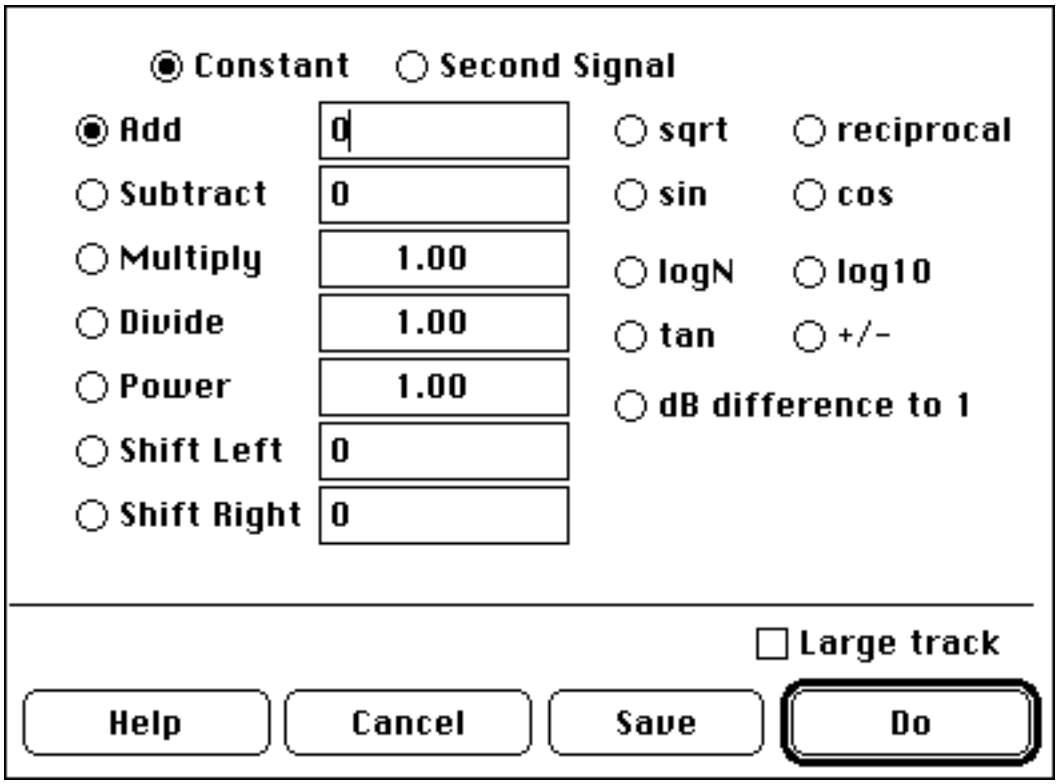
Window>Speed Palette
Fenêtre>Palette de vitesse
Fenster>Geschwindigkeitspalette

This command brings the Speed Palette to the front, if it is not already in front.

“Transform” Menus

Transform>Simple Transformation Setup...

Transformer>Options pour transformations simples...
Umsetzen>Optionen Einfache Umwandlungen...



Performs simple transformations to generate new signals from existing signals.

How To

See Tutorial 7.

General Information

- Some of these transformations can involve operations with either a *constant* or a *second signal*. For operation with two signals, select the first signal with the 1-button, and the second signal with the 2-button.
- If the operation is performed with a constant, type the constant value into the edit field and click on the radio button for “constant”.
- If the operation is performed with a second signal, click on the radio button for “Second Signal”. Constants shown in the edit fields are ignored with the Second Signal option.

- To transform a complete signal, make sure that the entire source signal is visible on the screen.
- To transform part of a signal, display just part of the signal to transform, or drag-select the target segment.
- In the case of two signals, the intersect between the signals is processed.
- Operations proceed by successive samples, not by points-in-time. Therefore, operations involving two signals with *different sampling rates* may provide unexpected results. Example: When adding a signal sampled at 10 kHz to a signal sampled at 20 kHz, samples representing different points in time are summed. If necessary, re-sample one of the two signals with the Complex Transformation in Transform>Complex Transformation.
- *Overflow:* Amplitude peaks of transformed signals can be in excess of the -32,766 to +32,766 range. Excessive values are top-and-bottom limited to the values +/-32,766.
- *Speed:* The speed of these operations varies mostly as a function of the numerical operation involved. Transcendental functions, multiplication and division are performed in floating point precision and are thus more time-consuming than addition and subtraction which use integer math. Left- and right-shift operations are highly efficient.
- *Transcendental Transformations:* Some of these operations can furnish results whose decimal portions are significant. Such values cannot be represented satisfactorily in the 16-integer domain. These operations are thus followed by a multiplication with an empirically derived constant. The constants used are:

OPERATION	NAME	CONSTANT
sqrt	square root	100
sin	sine	10000
tan	tangent	10000
cos	cosine	10000
logN	natural log	1000
log10	log10	1000
reciprocal	1/x	32767/x
dB difference	dB	100

Example:
sample x has a value of 22
transformed value = sqrt(x) * constant
transformed value = 4.69 * 100.0
transformed sample value = 469

Further, some of these transformations cannot operate on negative numbers (e.g., square root and log operations). Since users are generally interested in the transforming effect these operations can have on the signal, negative values are treated like "pseudo-positive" values. Negative values feeding into sqrt, logN and log10 are thus negated, then transformed and then re-negated.

Also, the operations sine, cosine and tangent are most valuable if they operate in the 0-±90 degree domain. Consequently, input values are mapped into that domain before being converted.

- For more information on numeric operations, consult a standard reference for the C language, e.g., Schildt (1987).

Operations

Add: Integer summation of samples of the selected signal and a constant, or the corresponding samples of a second signal. Permissible constants: -32,000 - +32,000.

Subtract: Integer subtraction of a constant, or samples of a second signal, from the corresponding samples of a selected signal. Permissible constants: -32,000 - +32,000.

Multiply: Floating point multiplication of samples of a selected signal with a constant, or with the corresponding samples of a second signal. Multiplication by a decimal value is possible (e.g., multiplication by -0.5). Permissible constants: -32,000.0 - +32,000.0. For fast multiplications by a power of 2, see "Shift Left".

Divide: Floating point division of samples of selected signal by a constant, or by samples of a second signal. Divisors of 0.0 are not legitimate (the operation results in a "missing value"). Division by a decimal value is possible (e.g., division by -0.5). Permissible constants: -32,000.0 - +32,000.0. For fast division by a power of 2, see "Shift Right".

Power: Exponentiation of samples of selected signal to the power of the listed constant, or to the power of a corresponding sample of a second signal. Permissible exponents: -20.0 - +20.0. *Caution:* The results of this operation easily exceed the ±32,766 range available with 16-bit format.

Shift Left: Bitwise left shift performed on a source signal to the extent indicated by the listed constant. Bits at the extreme left of the source samples are lost and newly created bits at the right have the value 0. Permissible constants: 1-16.

- Because of the way C stores signed integers, a left shift with constant 16 sets positively valued samples to 32,767 ("missing value") and the resulting signal is not visible. Samples with 0 are set to 0, and samples with negative values are set to -32767.
- Since a left shift is in effect a very fast multiplication by a given power of 2, this operation can be used advantageously in the place of

many multiplication and exponentiation operations. Example: to obtain multiplication by a factor of 2, left shift with a constant of 1, for multiplication by 4, left shift with 2, for multiplication by 8, left shift with 3, etc.

Shift Right: Bitwise right shift performed on a signal to the extent indicated by the listed constant or by the corresponding sample of a second signal. Note that bits at the extreme right of the source samples are lost and that the newly created bits at the left have the value 0. Permissible constants: 1-16.

- Because of the way C stores signed integers, a right shift with constant 16 sets all samples to 0 in the case of positive values or zero, and to negative 1 in the case of samples with negative values.
- Since a right shift is in effect a very fast division by a given power of 2, this operation can be used advantageously in the place of many division operations. Example: to obtain division by 2, right shift with a constant of 1 to divide by 4, right shift with 2, to divide by 8, right shift with 3, etc.

Sqrt: Samples in the destination signal will be (100 * square root) of corresponding samples in the source signal. Samples are negated, transformed and re-negated if ≤ 0.

Reciprocal: Samples in the destination signal will be a reciprocal (32,767 * 1/x) of corresponding samples in the source signal.

Log10: Samples in the destination signal will be the (1000 * log10) function of corresponding samples in the source signal, multiplied by 1000. Samples are negated, transformed and re-negated if ≤ 0.

LogN: Samples in the destination signal will be the natural logarithm of corresponding samples in the source signal, multiplied by 1000. Samples are negated, transformed and re-negated if ≤ 0.

Sin, Cos, Tan: Scaled samples in the destination signal will be the sine/cosine/tangent of corresponding samples in the source signal, multiplied by 10000.

- *Zero Baseline:* Before these operations are performed, the signal's offset bias is substracted from the signal value. This is done to prevent unnecessary bitwidth overflow and to provide a signal balanced around zero for the correct application of these transformations.

+ / -: Samples in the destination signal will be the opposite sign of corresponding samples in the source signal.

dB difference from 1: Samples are compared one by one to the value 1 and the dB difference is calculated by the formula:

```
diff = log10(abs(val)/1) * 20;
```

Please see Tutorial 7 for more details on amplitude calculation.

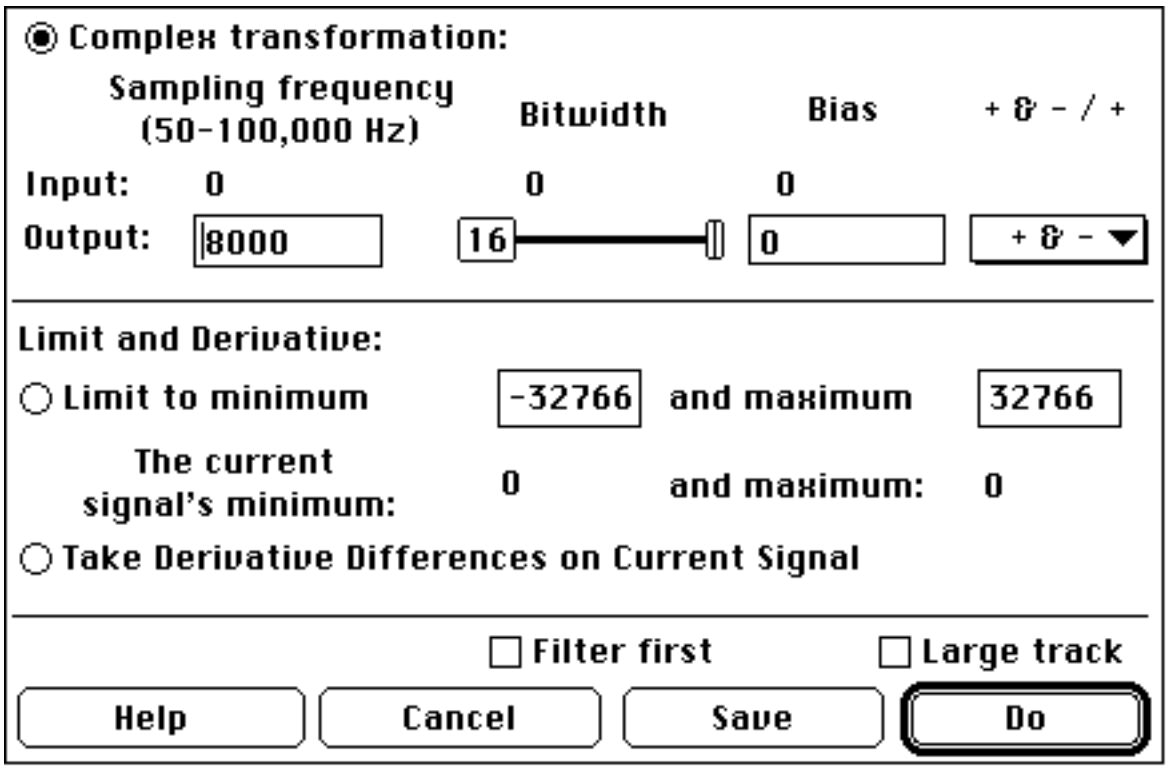
Transform>Simple Transformation

Transformer>Transformation simple
Umsetzen>Einfache Umwandlung

Performs a simple transformation in accordance with specifications in Simple Transformation Setup.

Transform>Complex Transformation Setup...

Transformer>Options pour transformations complexes...
Umsetzen>Optionen Komplexe Umwandlungen...



How To

See Tutorial 7.

Complex Transformation

This routine transforms signals according to four parameters, *sampling frequency*, *bitwidth*, *bias* and *numeric type*. Any of these can be modified or held stable.

- *Sampling frequency:* This parameter is modified by averaging plus removing, or by interpolating plus inserting samples at selected intervals (“downsampling” / “upsampling”, or “desampling” / “prosampling”). Example: A 11 kHz signal is converted into a 10 kHz signal by obtaining

the average of every 10th and 11th sample, by replacing the 10th sample by the average, and by removing the 11th sample. Or a 10 kHz signal is converted into a 11 kHz signal by interpolating between every 10th and 11th sample, and by inserting the interpolated value after the 10th sample. For fast operation, the calculation of which samples to remove or duplicate is performed in fixed-point math.

- *Bitwidth:* This parameter is modified by left- and right-shift operations. Example: A 16-bit signal is converted into an 8-bit signal by right-shifting each sample by 8 binary positions. Or an 8-bit signal is converted into a 12-bit signal by left-shifting each sample by 4 binary positions. The new binary positions created to the left or the right of the number are filled with zeroes.
- *Bias (or " offset bias"):* This parameter is modified by adding the specified number to the signal's average, as it was established when the signal is opened or recorded.
- *Numeric type:* Sample values can be either entirely in the positive domain (between 0 and 32,767) or can extend over both, positive and negative domains (between -32,767 and +32,767). Different digitizing devices use different numeric types, so your signals may be natively "unsigned" (entirely positive) or "signed" (positive and negative). (Remember also that signals become signed at input if you select "zero offset" in File>File Setup.)

When the numeric type is changed with the menu popup, an appropriate offset is suggested in the offset bias edit box. For signed output signals, the offset is 0. For unsigned output signals, the offset depends on the bitwidth. In the case of bitwidths 8-15, it is 1/2 of total range. In the case of bitwidth 16 it is always set to 0, due to range limits of a 16-bit signal.

- **CAUTION:** Upsampling from low sampling frequencies can create signals that easily exceed available memory.

Limit and Derivative Differences

- *Limit:* A signal can be limited ("clipped") to user-specified values. This can make weak signal portions appear more clearly.
- *Derivative Differences:* A signal can be differentiated by taking differences between succeeding samples. E.g.,

ORIGINAL	SIGNAL DERIVATIVE
400	
300	-100
50	-250
75	25

A derivative operation is often applied to enhance the high frequency components of a signal.

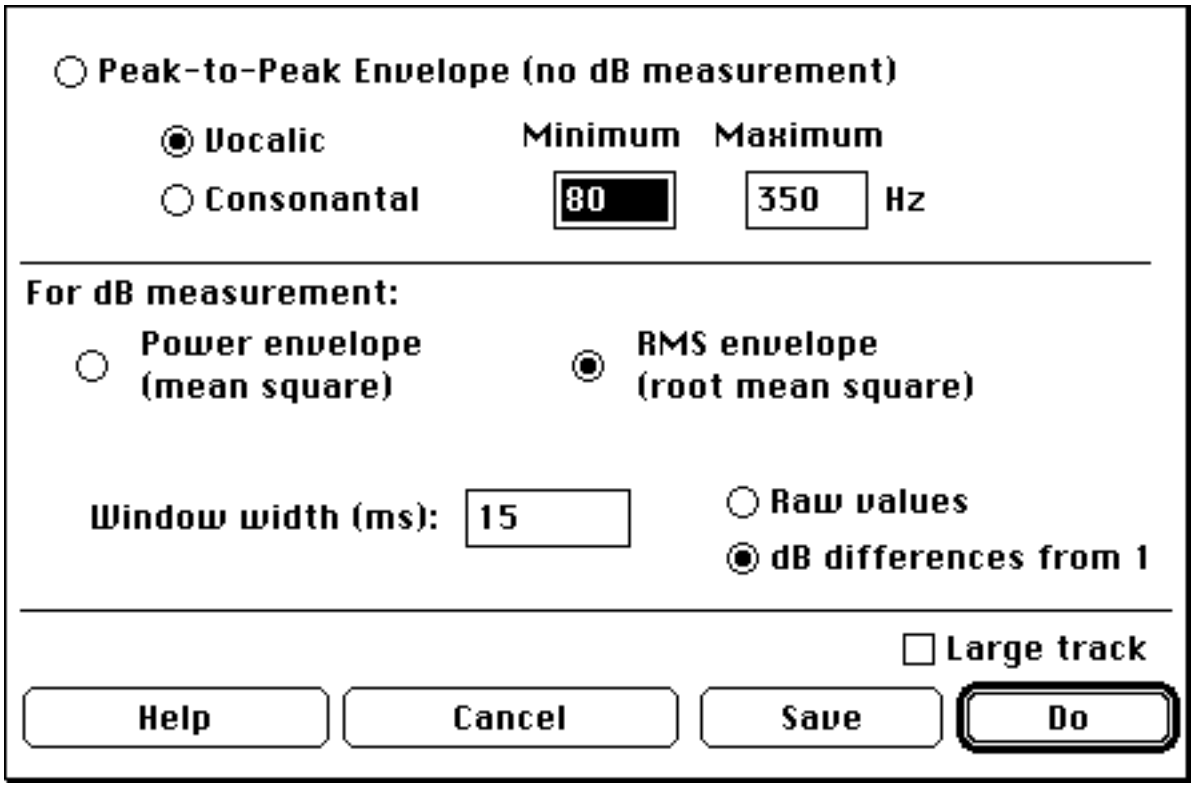
Transform>Complex Transformation

Transformer>Transformation complexe
Umsetzen>Komplexe Umwandlung

Initiates a transformation according to specifications in Transform>Complex Transformation Setup.

Transform>Envelope Setup...

Transformer>Options pour enveloppes d'amplitude...
Umsetzen>Optionen Amplitudenhülle...



Envelopes are typically used to investigate overall amplitude development in speech. Amplitude envelopes are differentiated by whether they admit of dB measurement or not.

How To

See Tutorial 7.

A Fast Envelope: The Peak-to-Peak Envelope (no dB measurement)

This routine picks out peaks in the positive domain and connects them to form a fast amplitude envelope. Envelope generation involves two passes over the signal. In the first pass, adjoining samples are averaged to reduce spurious peaks. In the second pass, as soon as a value greater than

the signal's mean value is found, an analysis window is examined for the window's maximum. Once the maximum is found, the routine jumps ahead by the time equivalent of the upper frequency bound (e.g., by 5 ms, if the upper bound is 200 Hz), establishes the next analysis window and finds the next local maximum. A local maximum "remains in effect", i.e., is graphically represented in successive pixels, till it is replaced by a new local maximum.

- *Vocalic:* This option tends to identify local maxima of the large glottal pulse found in vocalic portions of the signal. It thus imposes an envelope of optimal sensitivity for vowels.
- *Consonantal:* This option specifies a higher frequency band more relevant to consonantal activity.
- *Minimum and Maximum:* The minimum frequency controls the width of the analysis window and the maximum frequency controls the jump between the last found local maximum and the left edge of the next analysis window.

Power and RMS Envelopes (for dB measurement)

The power ("mean square") and the RMS ("root mean square") envelopes are closely related.

- A *power envelope* is calculated by entering into a sum the squared value of each sample of a given window (window width is specified in ms). This moves negative samples into the positive domain. The envelope is then drawn on the basis of the moving average (the "mean") of the window, as the window is moved, sample by sample, over the source signal. Hence this is a "mean square" [or "mean of squared values"] envelope.

In the case of signals with bitwidths greater than 9, the results of this operation can exceed the 16-bit numeric range. For this reason, results are further divided by a constant which depends on the source signal's bitwidth:

```
if (bitwidth > 13) div = 100 000;
else if (bitwidth > 9) div = 1 000;
else div = 1;
```

- *RMS envelopes* are calculated in the same manner, except that square roots are taken of the results of the mean square operation. Hence the name: "root mean square" (RMS) or "the root of the mean of the squares (in a given window)".

Although the taking of roots slows the routine down, the RMS envelope is often more informative than a power envelope, since the relation between weaker and stronger values is the same as in the original signal.

- It is possible to show either raw values resulting from the power or the RMS calculation, or dB differences between such raw values and the value 1 obtained by the formula:

```
diff_dB = log10(val/1) * 10; // for power envelope
diff_dB = log10(val/1) * 20; // for RMS
```

- To effect specific measurements of dB differences, specify “dB difference...” in Edit>Show Statistics Options. When Edit>Show Statistics (COMMAND-U) is requested, this calculates dB differences between between two features in the RMS or amplitude plot. dB differences are shown in the Edit Window.
- **CAUTION:** Most signals used in speech research are relatively ill-suited for amplitude analysis. Use such results with great care. For details, see Tutorial 7.

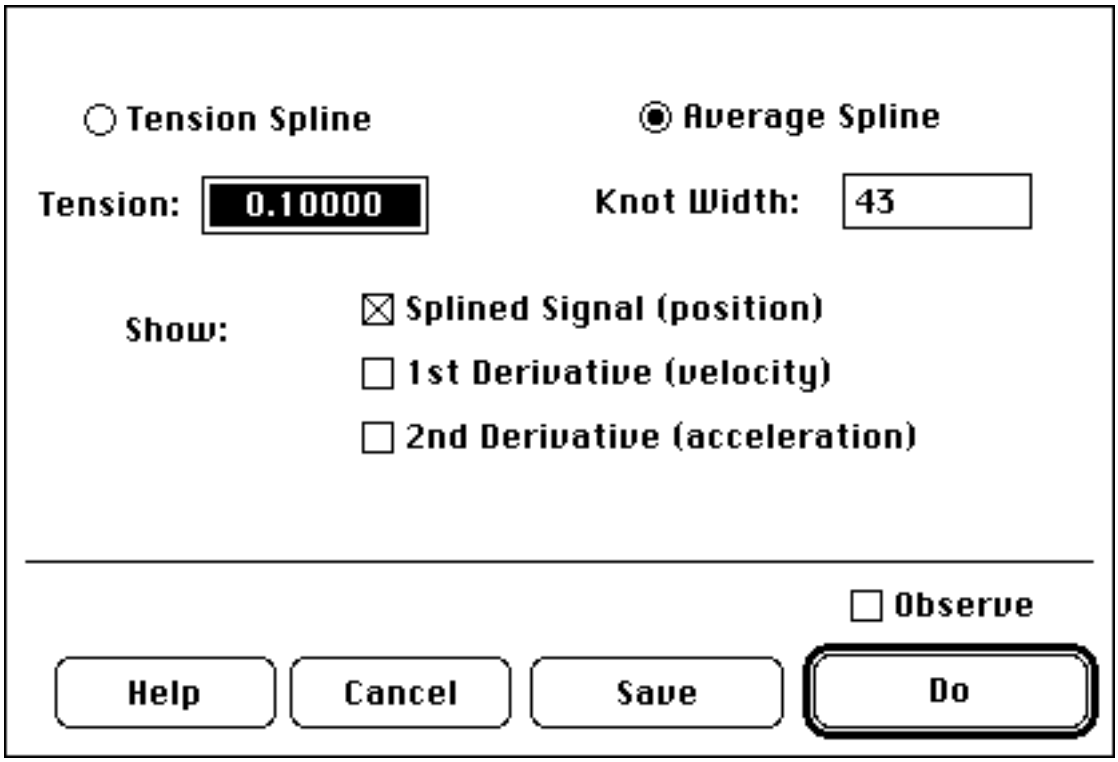
Transform>Envelope

Transformer>Enveloppe d’amplitude
Umsetzen>Amplitudenhülle

Initiates calculation of an amplitude envelope in accordance with specifications in Transform>Envelope Setup.

Transform>Spline Setup...

Transformer>Options pour courbes spline...
Umsetzen>Optionen Splineglättung...



Splines are part of a family of curve fitting techniques that smooth a signal while attempting to remain true to observed values. Splines do so by means of defining polynomial functions, which permits the easy extraction of a curve's first and second derivatives. In the case of movement ("kinematic") signals, the first derivative corresponds to *velocity* and the second derivative to *acceleration*. Two types of splines are available: a tension spline and an average spline.

How To

See Tutorial 7.

Tension Spline

The tension spline calculates a polynomial curve passing within a specified distance of each sample value. If the "tension" is great (i.e., close to 1.0), the curve has to pass closer to each sample value (and is thus less smooth) than if the tension is less (i.e., consideraly less than 1.0). A good tension to start with is 0.1. Permissible tensions: 0.00001 - 1.0. Permissible number of samples: 4 - 32,000. Upon entry into the tension spline routine,

the number of samples in the selected range is calculated. Tension spline calculation cannot be interrupted.

CAUTION: The algorithm uses six floating point matrices, and requires 10 bytes of calculation memory per sample per matrix. For 1000 samples, that makes 60 k of calculation memory. *There is no way around these massive memory requirements.* To avoid obtaining the "Insufficient memory" message, be sure you reserve enough calculation memory through Edit>Startup Options (also consider using virtual memory). Alternatively, check out the average spline routine which uses less memory.

Average Spline

The average spline is more efficient in memory use than the tension spline, since it first calculates adjoining averages (so-called "knots") and then connects the knots by curves described as polynomial functions. Minimum: 4 knots.

- *"Knot width"* corresponds to the width of the sampling window over which averages are calculated. With greater knot widths, the resulting curve is smoother, and with smaller knot widths it is rougher.
- *Calculation Memory:* The routine requires 85 k, plus 56 bytes per knot of calculation memory. To obtain the number of knots, divide the total number of samples by the knot width, truncate and add 1, e.g., a signal of 5000 samples at a knot width of 43 samples requires 117 knots. Permissible knot widths: 4 - 32,000. Permissible number of knots: 2 knots - 1 billion knots.
- *"Observe":* When this option is marked, intermediate results of the smoothed signal (derivative 0) are plotted. Turning this option off results in markedly faster spline calculation.

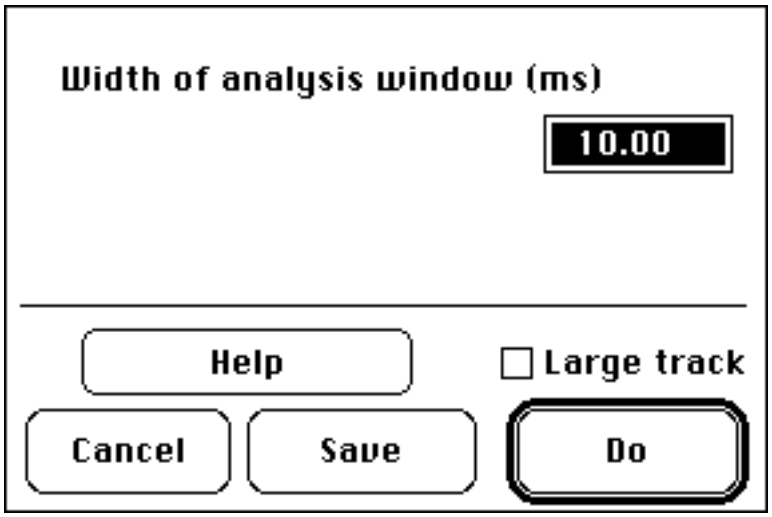
Transform>Spline

Transformer>Courbe spline
Umsetzen>Splineglättung

Initiates spline fitting in accordance with the specifications given in Transform>Spline Setup.

Transform>Zero Crossing Setup...

Transformer>Options pour passages par zéro...
Umsetzen>Optionen Nulldurchgänge...



The frequency of zero crossings is a widely-used index of frication in speech analysis. The highly irregular signals found in [f], [s], [ʃ] and other fricatives tend to cross the signal's resting value much more frequently than, for example, the regularly oscillating signal found in vowel portions of the speech signal. In Signalyze, the resting value is given as the linear mean of all samples.

How To

See Tutorial 7.

Specifications

- The *width* of the analysis window is given in ms. E.g., with a 10-ms window specification, zero-crossings are counted over adjoining 10-ms sliding windows (one window per sample position). The resulting count is multiplied by 100 to obtain crossings per second. Permissible widths for analysis window: 0.0001 - 1000.0 ms.
- **CAUTION:** Strongly *voiced* portions of speech carrying frication may be missed by the zero crossing routine, since voice oscillations deviate considerably from the signal's mean value. Zero crossings may thus be taken as a first index of frication, but should not be considered to be the definitive arbiters concerning the presence or absence of frication.

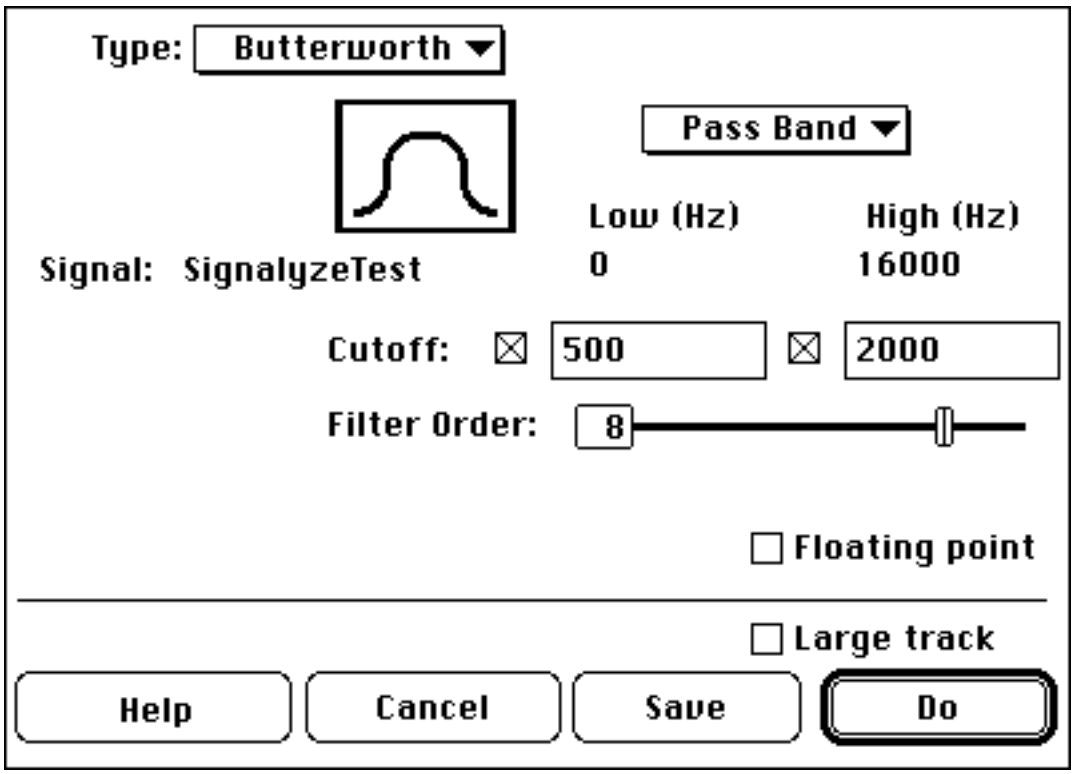
Transform>Zero Crossings

Transformer>Passages par zéro
Umsetzen>Nulldurchgänge

This command causes a zero crossing extraction to be performed on the selected signal as specified in Transform>Zero Crossing Setup. The readout is in Hz, or crossings per second.

Transform>Filter Setup...

Transformer>Options de filtrage...
Umsetzen>Optionen Filterung...



A filter changes the frequency characteristics of a signal. Typically, certain frequency components in the input signal are suppressed while other frequency components are preserved.

How To and More Explanations

See Tutorial 7.

FIR (Finite Impulse Response) Filters

- **Fixed 3-point Triangular Filter:** This filter calculates averages as follows: point 1 contributes 25%, point 2 contributes 50%, and point 3 contributes 25% of the value of the average. Because of its fixed 3-sample width, this filter is very fast. The filter uses a buffer in calculation memory as large as the signal itself, and can thus only be applied to relatively short signals. This filter is employed when "smoothing" is selected in the "Spectral Analysis Setup" dialog.
- **Fixed 7-point Hamming Filter:** This filter has smooth edges (somewhat like a normal distribution) designed to eliminate undesirable side effects. Points 1 and 7 contribute 2.34375%, points 2 and 6 contribute 8.59375%, points 3 and 5 contribute 23.4375% and point 4 contributes 31.25% to the average. (These are fast, shift bit math-adapted approximations to real hamming distribution values.) Because of its fixed 7-sample width and shift bit calculations, this filter is very fast. The filter uses a buffer in calculation memory as large as the signal itself, and can thus only be applied to relatively short signals.
- **Moving Average:** This filter has a square window shape of settable width. At narrow window widths, the abruptness of entering new values at their full strength may introduce undesirable side effects.

IIR (Infinite Impulse Response) Filters: Butterworth and Chebychev

- *Butterworth vs. Chebychev:* The main difference between the two filters relates to the steepness of the frequency cutoff. At the same order number, the cutoffs of Butterworth filters are less steep than those of Chebychev filters. At the same time, Butterworth filters introduce fewer negative side effects ("ripple effects") than Chebychev filters.
- Frequencies "passed" by Chebychev filters (i.e., those NOT suppressed) are admitted at *unequal levels of amplitude*, i.e., some frequencies are passed at the original amplitude, while at other frequencies, amplitude is reduced or increased. This "ripple" is measured in dB and can be controlled to some degree by the user. (Set "ripple" to low dB levels).
- *Higher Orders—Greater Steepness:* In both Butterworth and Chebychev filters, higher order numbers give greater steepness at the cutoff frequency—at the cost of some additional computing time. Also, at the same order number, Chebychev filters introduce greater deformations of the signal's phase characteristics than do Butterworth filters.

- *The Signalyze Implementation.* Signalyze implements only high-pass and low-pass IIR filters. Band-pass is simulated by first low-passing, then high-passing a signal. This takes only marginally longer than applying a real (and fairly complex) band-pass filter.
- *Floating point:* The Butterworth and Chebychev filters are implemented in three forms:
  — Floating point with co-processor
  — Floating point without co-processor
  — Fixed point

When the floating point option is selected and a co-processor is present, the program automatically switches to the appropriate routine, as long as the co-processor is enabled in Edit>Preferences. Please refer to explanations about co-processor use in the section on "Edit>Preferences".

- *Cautions About NOT Using Floating Point:* On computers not equipped with a co-processor, the fixed-point implementation is a great deal faster than the floating point implementation. Under most conditions, good filtering characteristics are preserved.

However with *high order numbers* (e.g., above 5), and/or with low *frequency cutoffs* (e.g., below 500 Hz), filter coefficients may at times attain very high numbers leading to fixed-point math overflow. This results in a seriously degraded signal. Under those circumstances, a lower order number or the floating point implementation of the filter should be selected. At the same order number, Butterworth filters show less tendency toward overflow than do Chebychev filters.

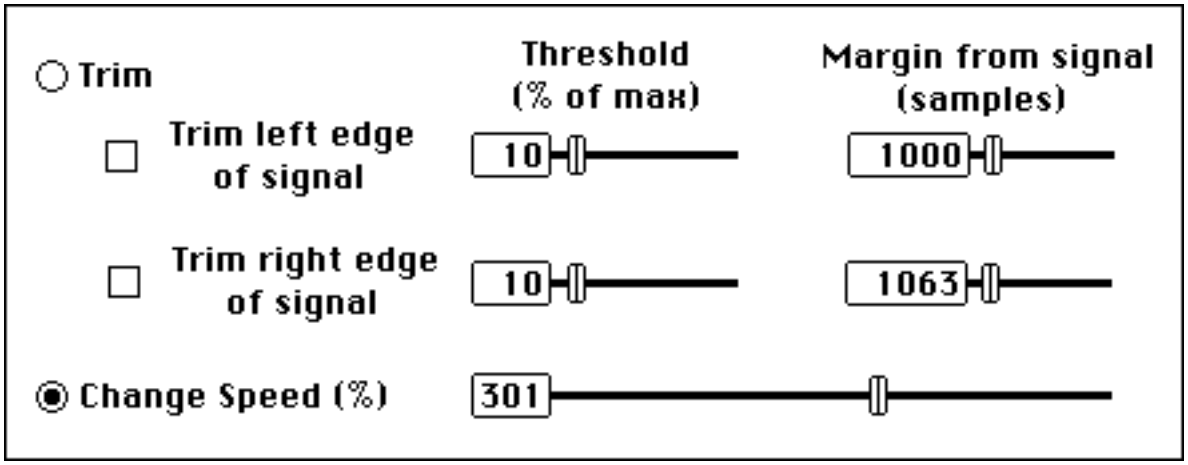
Transform>Filter

Transformer>Filtrage
Umsetzen>Filter

Initiates signal filtering according to specifications in Transform>Filter Setup.

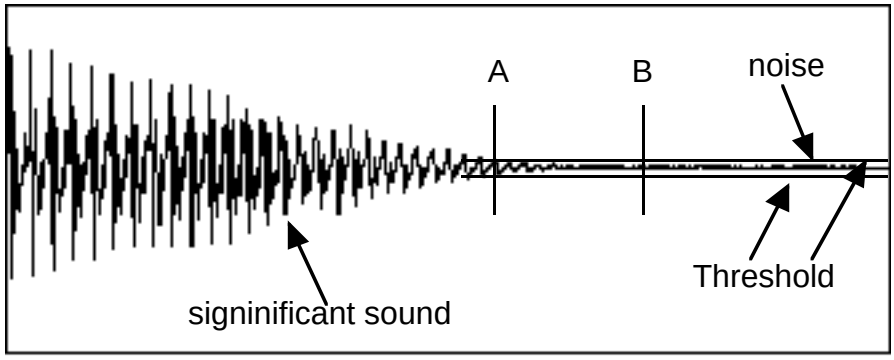
Transform>AutoEdit Setup

Transformer>Options d'édition automatique
Umsetzen>Automatische Bearbeitung



Trimming

When you have a signals with quiet edges, it saves a lot of disk space to be able to trim off the edges. This routine lets you do that semi-automatically.



• *Operation:* The threshold percentage establishes two lines around the signal's zero-line. This percentage is set to discriminate between background noise found in the "silent" portion of the signal and a significant sound level. The intersection between this threshold and a sound level exceeding the threshold is considered to be onset **A**. Trimming occurs at a point removed from onset **A** by the set number of marginal samples, at point **B**.

Changing Speed

This routine uses the Signalyze speed modification techniques (also used in the Speed Palette) to create speeded-up or slowed-down versions of a speech signals.

- *Operation:* The routine attempts to identify the large positive and negative-going peaks representative of the fundamental frequency. It then identifies points roughly midway between these peaks. When the signal is speeded up, signal portions between these midpoints are removed at regular intervals. When the signal is slowed down, such signal portions are duplicated as many times as required. Transitions between removed or duplicated portions are smoothed by straight “connecting lines”.
- **Caution 1:** The removal or insertion of sound segments nearly inevitably introduces “transition noise”. The amplitude and annoyance level of this noise depends on the specific portion of the signal where the transition occurs.
- **Caution 2:** With some voices, peak detection is particularly difficult. No pretense is made here that this routine will handle all voices correctly. However, it has been found that the routine works reasonably well with nearly all voices that have been tested.
- **Caution 3:** Professional speed modification programs rarely exceed a speed-up beyond 75% and a slow-down beyond 150%. Even though it is possible to go well beyond these limits with the Signalyze routine, transition noise may increase to such a degree that exceeding these limits cannot be recommended without caution.

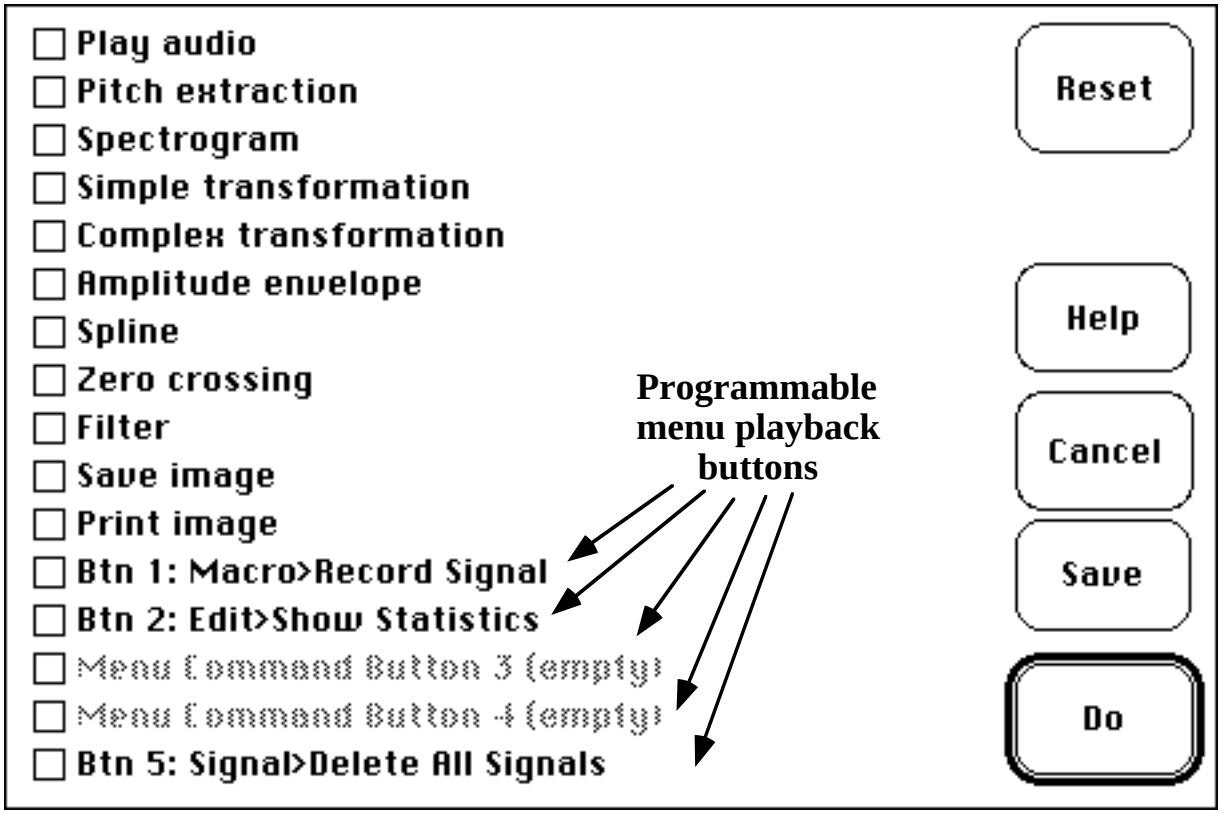
Transform>AutoEdit

Transformer>Edition automatique
Umsetzen>Automatische Bearbeitung

Effect a signal modification according to the specifications in Transform>AutoEdit Setup.

"Macro" Menus

Macro>Sequential Macro Setup...
Macro>Options de macro...
Makro>Optionen Serienmakro...



This dialog specifies a series of operations to occur in sequence. This facility complements the programmable buttons that can store menu commands (top of the General Control Panel).

How To
See Tutorial 8.

Specifications
- *Select processes* by clicking buttons in the desired sequence. The sequence number appears to the left of the process. Select again to eliminate a process from the sequence.
- *Execute* the macro by clicking on "Do".
- *"Reset"* eliminates all present selections and lets you start a new selection process.

- *"Save"* stores the present set of macro options for later execution via the menu command Macro>Play Sequential Macro (COMMAND-4).
- All processes apply to the currently *selected signal* as they are defined in the various setup dialogs.
- Use the option "New signal after transformation" (in Edit>Preferences) to program *chains of sequential processes*.
- Use the option "Original signal after transformation" (in Edit>Preferences) for *pitch extractions, spectrograms and sound playback*. Otherwise second or later processes may not apply to the right signal.
- You can store Menu Playback Buttons (numbered buttons at the top of the General Control Panel) for even greater convenience. When the Sequential Macro is executed the menu actions stored in these buttons are performed in the requested sequence.

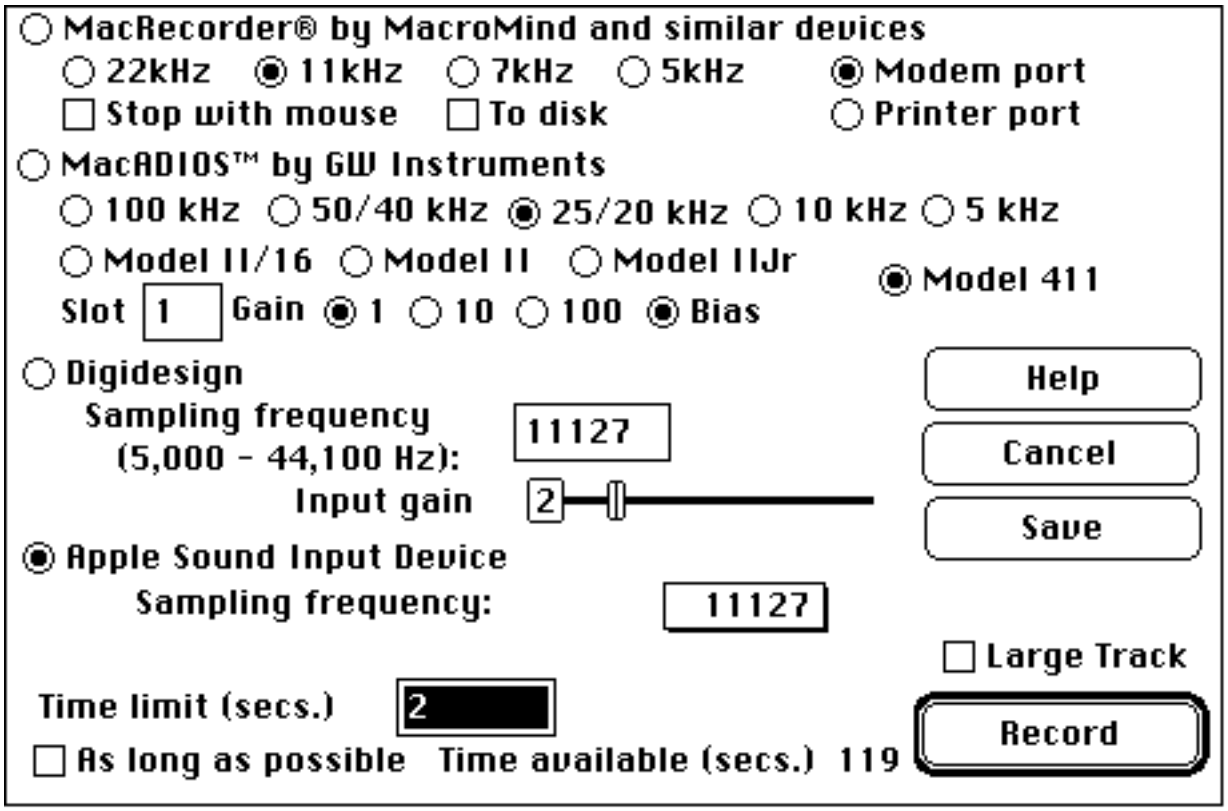
Macro>Play Sequential Macro ⌘4
Macro>Jouer macro séquentiel ⌘4
Makro>Serienmakro ⌘4

Executes the set of operations specified in Macro>Macro Setup.

Macro>Record Setup... ⌘2

Macro>Options de numérisation... ⌘2

Macro>Optionen Direkteingabe... ⌘2



This dialog controls the operations of analog-digital conversion ("digitization") via the converters supported by Signalyze.

How To

See Tutorial 3.

Caution: No guarantee is given for the correct operation of ***any of these routines other than the "Apple Sound Input Device" routine***. Specifically, recent sound devices or recent computers may not be compatible with the manufacturer-specific routines provided here.

If you encounter difficulties of this sort, please record with the manufacturer's own input program and play back via the built-in Apple sound output. Alternatively, place the manufacturer's Apple-compatible audio driver (if available) into your System folder and record via the "Apple Sound Input Routine Device" option.

No attempt will be made to "fix" manufacturer-specific input routines. Also please be advised that version 4.0 of Signalyze will no longer support any manufacturer-specific sound input. It will exclusively support drivers operating via the "Apple Sound Input Device" option.

Specifications

MacRecorder® and Similar Devices

MacroMind, Impulse and a number of small manufacturers distribute or have distributed devices that work essentially in the same manner. These devices may provide input compatible with Signalyze under this option. Code to read these devices has been adapted from a public, but copyrighted C-code project by the "Sid Trio".

- Specify the *sampling rate.*
- Specify the *modem/printer port* according to the serial outlet into which you've plugged the digitizing unit.
- **CAUTION:** The digitizing unit may not work if you also operate background programs (typically so-called "INITs") that regularly look at the modem or printer port. This concerns especially print spooler and FAX receiving programs. Disable conflicting background programs before you plug the digitizing unit into the modem or printer port. Note, however, that you *can* keep a background program going on one port while reading the digitizing unit at the other port (e.g., a FAX receiving INIT reading the modem port can be kept running while working with the sound digitizing unit via the printer port).
- *Stop with Mouse:* Mouse movements interrupt recording. Works with all recent Macs, but not the MacPlus.
- *Destination:* If the "To Disk" checkbox is selected, the routine records the input to disk continuously. Modern disks are generally fast enough to store the data as fast as it arrives. With slower disks, serious interruptions in sound recording may occur. If the checkbox is deselected, the routine records to RAM (signal memory).
- *Clipping:* To cut off input device settling, the initial 1000 samples are discarded from the recorded stretch of speech. However, the user-specified time limit is respected.

MacADIOS Devices

GW Instruments have put a series of 12- and 16-bit A/D-D/A devices on the market. Interface code for these devices is by GW Instruments and is used by permission.

- *Sampling rate:* Select the rate that your device supports.
- *Card or device:* Select the appropriate I/O device.
- *Slot, gain, bias:* In the case of the MacADIOS boards (not the Model 411), set the slot number, the input gain, and bias. Determine the slot number by visual inspection. The input gain can initially be set to "1" and can be increased as needed. The bias should be set to "on". It controls both input and output bias (i.e., the offset between signal averages required for Signalyze and for MacADIOS boards). Bias can be deselected if the sound is scratchy and excessively noisy.

- *Stopping recording:* The MacADIOS drivers used in Signalyze are "one-shot" algorithms that cannot be interrupted.
- *Destination:* The routine records only to RAM (signal memory).
- *Clipping:* No clipping of initial signal input is required prior to calling MacADIOS devices with GW Instrument drivers. However, the drivers themselves may perform some clipping of the initial signal segments.

Digidesign Devices

Digidesign Inc. sells a number of DSP boards for sound in- and output. A DSP board differs from an A/D-D/A board in the sense that it is built around a DSP ("Digital Signal Processing") chip that offers functions in addition to analog-to-digital/digital-to-analog conversion (e.g., FFT, filtering). The interface code employed in Signalyze has been written by Digidesign and is used by permission. It has been tested with the AudioMedia, AudioMedia II and Sound Accelerator boards, but it may work with other Digidesign devices as well.

- *Sampling frequency and Hardware Setup:* Any sampling frequency from 5 - 44.1 kHz is acceptable. The board samples at 44.1 kHz natively or can be set to sample at 44.1 kHz via the hardware setup found in the AudioMedia or the Sound Designer II program. The Signalyze input routines down-sample this input to the user-specified frequency. While in the hardware setup, also make sure that your AudioMedia board uses internal sync and that your Sound Accelerator board addresses the correct external A/D device.
- *Gain:* On AudioMedia boards, you can set gain with the slider. On the Sound Accelerator, set gain on the external input device.
- *Stopping input:* Click on mouse.
- *Destination:* The routine records only to RAM (signal memory).
- *Clipping:* To cut off initial input device settling, the initial 500 samples are discarded from the recorded stretch of speech. However, the user-specified time limit is respected.

Apple Input Devices

In System 7, Apple defined a input standard for its own as well as for third party sound input devices. The standard can handle 16-bit as well as 8-bit devices. For details, see "Summary of Version 3.0 Additions" at the beginning of this manual, and Tutorial 3.

- An appropriate sound *input driver* must be in the Extensions folder for this type of input to work right.
- *Setup:* Select the input driver through the "Sound" Control Panel accessory. Test the device through the accessory. If it works there, it should also work in Signalyze. If it doesn't work correctly, remove potentially conflicting sound input drivers from the Extensions folder. If it still does not work, please contact the manufacturer.

- *Stopping input:* The present interface code is written for "one-shot" recording and does not permit stopping in mid-course.
- *Automatic Gain Control (AGC):* Some input devices (such as Apple's microphone input driver) optionally control input gain automatically. This introduces amplitude modulations into the signal that are unwelcome in research applications. The Signalyze input routine disables AGC before recording.
- *Destination:* The routine records only to RAM (signal memory).
- *Clipping:* To cut off initial input device settling, the initial 500 samples are discarded from the recorded stretch of speech. This gets rid the digitizer's settling period. At the same time, the user-specified time limit is respected.

Other Specifications

- *Save:* Causes the settings to be stored as the dialog box closes.
- *Record:* Causes the settings to be stored and digitizing to be initiated as the dialog box closes.

A frequent customer inquiry is whether Signalyze will in the future support a series of additional A/D-D/A devices. The answer is: "not specifically". Please recall that Signalyze is above all an analysis program, that A/D-D/A programming is *exceedingly tedious and unpredictable* because manufacturers keep changing specifications and adding new, incompatible products. By far most Signalyze customers have now gone the Apple-compatible, the Digidesign and MacAdios routes, and above all, Apple has now defined a standard input mode fully supported by Signalyze. For sound input, it is clearly the manufacturer's responsibility to provide System 7-compatible input drivers.

Macro>Record Signal ⌘3

Macro>Numériser ⌘3

Makro>Direkteingabe ⌘3

Initiates analog-to-digital (A/D) conversion ("digitization") as specified in Macro>Record Setup.

- Good mike technique is essential to getting satisfactory A/D input. The idea is to record signals that fill the available range, yet show no "clipping". Clipped signals are basically useless for spectral analysis. Keep the mike a bit below or to one side of the mouth. If you speak right into the mike, plosive sounds provoke excessive impulse responses.

Technical Information on Signalyze

In this section, a number of technical issues about Signalyze are discussed. Some of the ergonomic and program design considerations that underlie the development of the program are sketched, file formats are described, customization is discussed, some of the most common questions about Signalyze are answered, and the license agreement is reproduced. The appendix lists abbreviations used in the program as well as the index to the Manual.

Ergonomic Considerations

Apple Macintosh Interface Guidelines

In order to simplify the learning process for new users, the program has been written with strict adherence to the Apple Human Interface Guidelines. Practically as a matter of professional pride (since the author is a professor of computer science), every aspect of the program has been examined for its simplicity, its clarity, and the logical extension of interface concepts used in other common Macintosh programs. To make learning transfer as easy as possible, several programs have served as basis of comparison and inspiration, especially text processing, graphics and spreadsheet programs. As a result, Signalyze should be quite intuitive in the way it presents and analyzes speech signals.

The Source of Complexity

Having said that, it remains that Signalyze is not a simplistic program, neither from the programmer's nor from the user's perspective. Enormous data sets must be quickly displayed and analyzed. Also, time synchronicity must be maintained over great differences in sampling rate and zoom. Finally, there are few programs that in- and output as many different data types—not only 8- and 16-bit signals of various formats, but also TEXT and PICT files. Not to mention that Signalyze handles digitizing and sound reproduction in four different physical environments.

The user is challenged as well. What are all the buttons for? How do I zoom the signal? Where do I read off the dB? What's an RMS envelope? Who's ever heard of splines? And is "cepstrum" simply a misspelling of "spectrum"? It's been said with justification that Signalyze requires an uncommonly steep learning curve of its users.

However, there's a method to the madness. The principle that's been followed is *to make the program as easy to use as possible, but not to shy away from complexity or novelty when the subject matter requires it.*

A good example is the arrangement of signal displays. SoundEdit users have come to expect a new window for each signal and are surprised to find several signals in the same window. But why, really? Spreadsheet programs habitually present related data in multiple, parallel tracks. Once the importance of synchronicity and of ease of comparison between several signals is understood, the extension of the window concept to multiple signals arranged in "cards" becomes nearly inescapable. And in fact, it can be argued that Signalyze's stable Signal Window incorporates much better the "perceived stability" concept of the Apple Human Interface Guidelines[1] than does the usual jungle of open signal windows, characteristic of other signal analysis programs on the Mac.

Similarly, the many advantages of 16-bit signals must be fully explained, since most users are used to the traditional Macintosh 8-bit sound environment. But once the importance of better sound reproduction and improved spectral analysis is understood, the user generally becomes more willing to pay the price in extra memory requests and in additional delays for sound reproduction.

In short, Signalyze will not shield you from a learning effort, but it will try to make the learning pay off for you. Speech signal analysis has evolved greatly from the time that we printed up spectrograms and measured spectral features on electrostatic paper. Signalyze is there to help users take systematic and concrete steps towards a more sophisticated approach to speech signal analysis.

Signal Window Layout

An ergonomic window and palette layout can save the user valuable time. The present signal window layout was based on considerations of available screen space, functional organization, as well as of minimizing of mouse travel. Several layouts were tried before the present design was adopted. *General functions* were grouped in a first button palette in the upper right hand corner, close to the menu bar. *Signal-specific functions* were arranged in a second set of palettes to the left of their respective signals.

---

[1] Apple (1987). Human Interface Guidelines: The Apple Desktop Interface, p. 8.

In addition to being intuitively functional, this arrangement permits easy resizing with larger monitor screens. The right margin of the signal window extends to give more display space, and additional signal tracks can be accommodated at the bottom of the window.

Displays, Signals and "Cards"

In a philosophy more akin to that of spreadsheet programs and HyperCard than to that of traditional text editing programs, multiple signals appear on a single window (a "card"). Any signal can be rendered visible by jumping to its card, or by moving it to an available display track.

The card concept has a number of advantages over the window-per-signal design:

- In a multiple signal analyzer, it is important to show related signals close to each other. Signalyze lets you rearrange your displays and lets you show related displays as close to each other as possible.
- Signals recorded at different sampling rates, as well as their spectrograms, are always displayed with the correct time scale.
- No screen space is wasted on a window frame and on scroll buttons around each signal.
- Signals remain in memory and can be re-displayed at will, even though they may be temporarily eliminated from all displays.
- Window switching is minimized and "perceived stability" is enhanced.
- The conceptual difference between Signal Window, Text Window and Plot Window is more evident and visual clutter is reduced.

Cursor Manipulation

There is only one vertical line cursor per signal. This incorporates a conscious ergonomic design decision taken after observation of numerous users of signal analysis software on a variety of computers. It was found that with programs that have two cursors, a high proportion of users—including the author—would generally forget which cursor was "active". As a result, users often click in the wrong places and then have to correct their clicking errors. On computers that use multi-button pointing devices, it was found that right and left buttons were confused by a fairly high percentage of users, especially with some fatigue.

It is for largely the same reasons that Apple decided on a single-button mouse when the Macintosh was developed. It was found that for a large number of users, a single button, activated once or twice, was less confusing than activating one of several closely-spaced buttons just once. Selection processes in text or graphic material thus typically involve the blackening of a selection region by button-and-drag operations,

principles that have been copied to the signal selection techniques in Signalyze.

Interactivity and Speed

Signalyze was developed from the bottom up as an interactive program, not as an off-line batch processor. Since the user interacts directly with the signal and its spectrographic and numeric information, speed of operation is of much greater importance than in traditional signal analysis programs. Operational and display speed have thus been one of the prime preoccupations in the design and construction of this program.

Signalyze *signal displays* are quite fast, among other things because they use bit mapping (called "blitting" in computer lingo). In this drawing mode, the signal image is first drawn into a special buffer memory, and is then copied practically instantaneously to the screen, even faster than a MacPaint image. (The Macintosh operating system uses the same approach for redrawing parts of the screen after menus are released). For drawing signals, Signalyze thus does not use the slower line drawing mode available through the Macintosh Toolbox. Also, as much as possible, Signalyze accelerates signal display through the use of integer arithmetic, bit shift calculations and register variables.

As fast as it is, signal display in Signalyze is just a bit slower than the displays of some other Macintosh applications, notably those of SoundEdit. The reason lies in Signalyze's exceptionally flexible multi-channel, variable sampling rate, variable zoom, variable amplitude, 2-16-bit conception. Each plot requires complex calculations of pixel location, sampling rate, zoom setting and amplitude magnification, and that separately for each signal. The program thus spends a fair bit of time on calculating what is to be displayed where. By contrast, 8-bit programs tied to specific input devices, and implementing relatively limited zoom manipulation, such as SoundEdit, benefit from a number of simplifying assumptions concerning the relationship between sampling rate, zoom setting, maximum amplitude and signal display. Further, 8-bit signals are in many respects inherently faster to manipulate than 16-bit signals.

Users of Classic-type Macs may wish to try out Signalyze on the faster Macintosh II-types of computer. Because of greater CPU speed, these machines can eliminate most of the small wait periods that the Macintosh Plus and SE computers impose on display operations. For true excitement, try Signalyze on a hyper-accelerated Power Macintosh…

Portable and PowerBook Adaptations

Signalyze incorporates special adaptations for use on Apple's portable computers. Specifically, the serial port is powered up during sound input, and the sleep feature is disabled during processes that take a certain amount of time.

RISC Processor Adaptations

Special adaptations to the RISC processors will be incorporated over the next few years to accelerate time-critical routines. Despite these adaptations, Version 3.0 will remain a "universal" application that will run on essentially the entire range of Macintoshes.

Programmability

Another major adaptation will be the introduction of scripting. When this extension is completed, Version 3.0 users will receive a free update.

Memory Management Considerations

Since signal analysis is by its very nature a memory-and-processor intensive application, it is assumed (and recommended) that Signalyze users give high priority to this program. On a Macintosh Plus or SE, Signalyze commands practically all the machine's resources, and on more powerful Macintoshes, Signalyze must be allocated a fair bit of memory and processor time to operate, otherwise its performance will be lackluster.

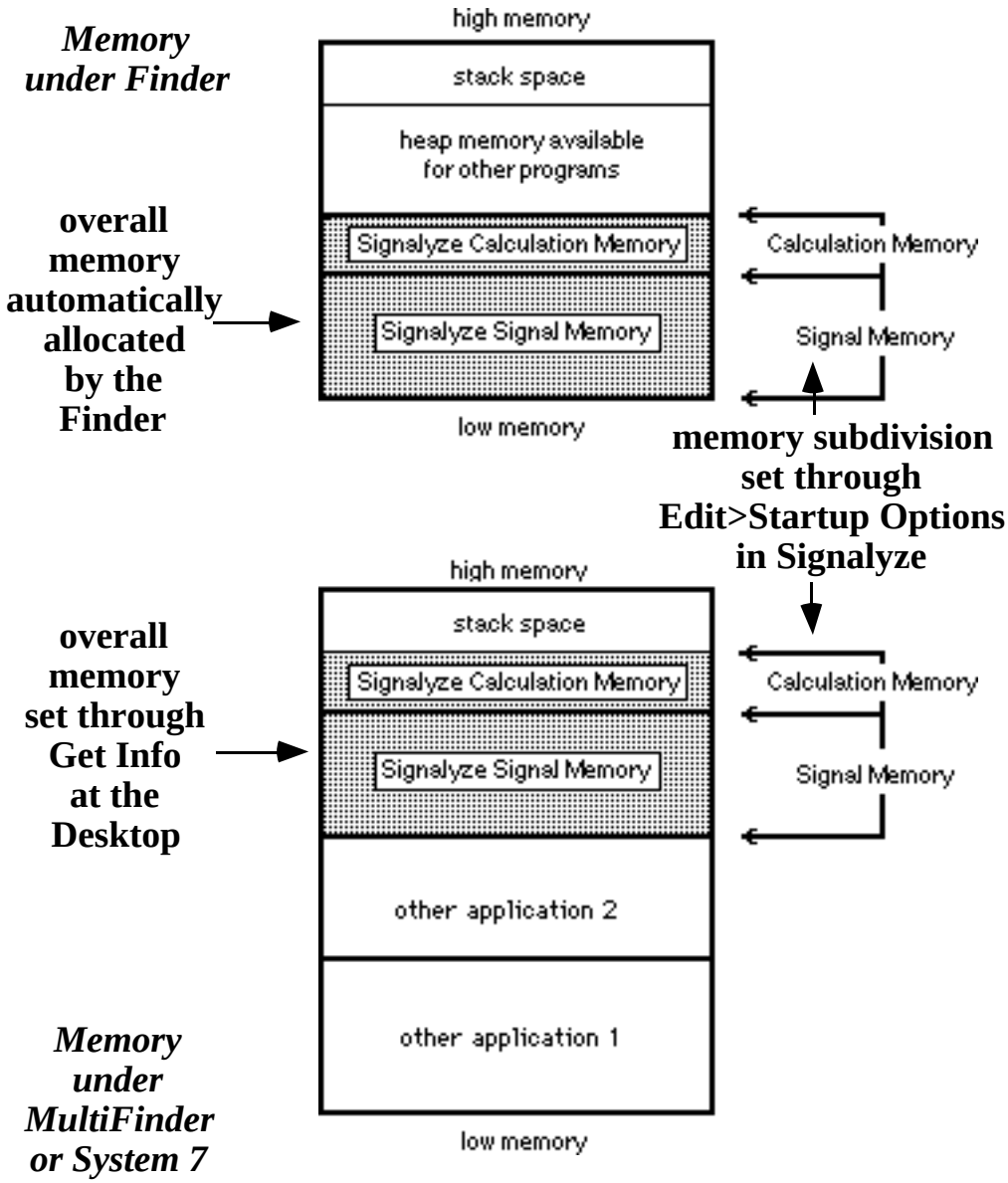
Memory Allocation

Signalyze requests most needed memory space close to the beginning of the program. This involves a massive block of non-relocatable memory in which signals are stored, and another block of assignable memory where calculations are performed. Under System 7, you can set the program's *overall* memory request by filling in the memory request in the Get Info dialog obtained at the desktop level.

Signal memory management is largely under the control of Signalyze. The application's signal and calculation memory block sizes are specified in Edit>Startup Options. At the onset of the program, Signalyze defines its global structures, increases the System's stack space, and reserves several off-screen signal buffers and a text buffer. Signalyze then attempts to gain control of as much unrelocatable memory as is specified for combined signal and calculation memory, using ResrvMem(). If a high number is specified (e.g., 8 Megabytes), ResrvMem() will do its best to find as much contiguous memory as possible. Within the identified contiguous memory space, a specified block is set aside for Signalyze *calculation memory*. Free memory

remaining in the identified memory space after this operation can be used for storing signals and is called *signal memory*.

A specified, non-moveable block the size of signal memory is then requested from the Macintosh memory manager via NewPtr(). If less than 50 k of memory is available, a message is generated and the program exits. If the requested free memory block is *smaller* than available contiguous memory, more memory is left for calculation memory, desk accessories (under Finder), and spoolers. The size of allocated and presently available *signal memory* can be verified through >About Signalyze. Also given in the same dialog box is the momentarily available amount of *calculation memory*.



Signalyze memory allocation under different systems.

Tight Ship: When using desk accessories under Finder and System 6, foresee a safety margin of minimally 200 k of calculation memory at all times. Many calculations (especially spline calculations and signal printing) require a fair bit of memory, and you increase the chance of unpredictable crashes if you try to fit too many programs into too little memory.

Why an Unrelocatable Memory Block for Signal Storage?

Some words are in order concerning Signalyze's design of requesting a large, contiguous and unrelocatable block of memory for signal space, rather than requesting memory blocks as signals are loaded in or generated. In many Macintosh applications, memory allocation is handled more flexibly through calls to the rather sophisticated Macintosh memory manager as the application is being used.

The reservation of a large "fixed block" of memory speeds up the loading and generating of long signal files. Otherwise blocks of unrelocatable memory would have to be requested from the memory manager each time a new signal is loaded or produced, leading to a fragmented and inefficient memory space. "Insufficient calculation memory" messages would become even more frequent than they are now.

Under the current arrangement, subsequent signals are loaded into contiguous memory locations. As signals are deleted, the "gaps" are closed to free a maximum amount of free memory. Also, the entire signal space is rearranged during "cut", "clear" and "paste" operations, imposing minor penalties in time for signal edit operations performed near the beginning of a fully loaded signal memory. The present design therefore favors plot, signal reproduction and transformation operations over signal edit operations, which is in tune with the main purpose envisaged for the program.

Signal File Formats

Although Signalyze supports a wide range of file formats for data files, it is suggested that as much as possible, users should translate their signals to the Signalyze 12/16-bit file format or to AIFF format. Please see Tutorial 8 for a detailed list of reasons why to adopt these formats.

Signalyze formats are hereby declared to be in the Public Domain. *The file formats used in some other programs are potentially subject to copyright claims.*

12 / 16-bit Formats

Signalyze Format
Company/Distributor: InfoSignal Inc./public
File type: DATA, Creator: BWSW
The header's first two bytes, or the extension field, are reserved for future use. Signalyze Versions 1.0, 2.0 and 3.0 place 0 in this structure field.
The sRate field is a *signed short.* The numbers 1 through 32767 are used to designate Hz, and the numbers -1 through -32766 are used to designate kHz. Examples: 100 designates 100 Hz, 32000 designates 32000 Hz, -32 designates 32 kHz and -32000 designates 32000 kHz, or 32 MHz.
Data is given as *signed ints,* where the ideal resting level is 0, the most negative amplitude is -32,766 and the most positive amplitude is 32,766. The value 32,767 is reserved for "missing value". Samples coded with this value are counted as samples (and thus take up time and memory), but are not shown on the screen.

```
typedef struct SignalyzeHeader {
    short       extension;      2 bytes, reserved,
                                set to 0 in version 1.0.
    long        fileLen;        4 bytes, sound data length
    short       sRate;          2 bytes, sampling rate
                                (1 through 32767
                                and -32767 through -1)
} SignalyzeHeader;
```

Audio Interchange File Format (AIFF):
Company/Distributor: Public
File type: AIFF, Creator: *[varies]*
Audio IFF files conform to the EA IFF 85 standard. They consist of a number of "chunks". Each chunk is made up of a header, containing the chunk's ID and data size, and an array with the actual data. The only required chunks are the Form Chunk which is used as a container for all other ("local") chunks, the Common Chunk with its information about number of channels, number of sample frames, bit width and sampling frequency, and the Sound Chunk which contains the actual signal data. On reading, Signalyze ignores all but these three chunks.
As an interchange format, AIFF is very flexible regarding signal specifications. It can store mono or multi-channel signals, together with various additional information. The allowed bit width range is 1 to 32, and there are no limitations on sampling frequency (as long as it is the same for all channels).
FileConverter converts an *n*-channel AIFF file into *n* Signalyze files or a Signalyze file into a 1-channel AIFF file.

A more detailed description of the AIFF format is given in chapter 22 of "Inside Macintosh, Volume V" (Apple Computer, 1988).

```
typedef struct {
        OSType          ckID;                   /* 'FORM' */
        long            ckSize;
        OSType          formType;               /* 'AIFF' */
        char            chunks[];
} FORMchunk;

typedef struct {
        ID              ckID;                   /* 'COMM' */
        long            ckSize;                 /* 18 */
        short           numChannels;
        unsigned long   numSampleFrames;
        short           sampleSize;
        double          sampleRate;
} CommonChunk;

typedef struct {
        ID              ckID;                   /* 'SSND' */
        long            ckSize;
        unsigned long   offset;
        unsigned long   blockSize;
        unsigned char   soundData[];
} SoundDataChunk;
```

MacSpeech Lab Format

Company/Distributor: GW Instruments

File type: DATA, Creator: GWI

The header's first two bytes, or the extension field, is reserved by GW Instruments. Most MacSpeech Lab files contain 1 in this structure field. The number in this field has no effect on Signalyze.

The sRate field is an *short*. Sampling frequency is coded according to the recording frequencies available through MacADIOS A/D and D/A converters. sRate equivalences: 16 — 83,332 Hz, 8 — 41,666 Hz, 4 — 20,833 Hz, 2 — 10,417 Hz, and 1 — 5,208 Hz.

Data furnished by MacADIOS 12-bit A/D converters is given as *unsigned shorts*, where the ideal resting level is 2048, the most negative amplitude is 0 and the most positive amplitude is 4096.

Data furnished by MacADIOS 16-bit A/D converters is given as *signed shorts*, where the ideal resting level is 0, the most negative amplitude is -32,766 and the most positive amplitude is +32,766.

```
typedef struct MacSpLabHeader {
        short           extension;              2 bytes, reserved
        long            fileLen;                4 bytes, sound data length
        short           sRate;                  2 bytes, sampling rate
```

(coded according to
MacADIOS standard)

} MacSpLabHeader ;

SoundDesigner II Format (Sound Accelerator, AudioMedia)
Company/Distributor: Digidesign Inc.
File type: Sd2f, Creator: Sd2a
SoundDesigner II files maintain three string resources called "sample-size" (STR resource 1000), "sample-rate" (STR resource 1001) and "channels" (STR resource 1002). Pertinent information in these resources is given in ASCII Text form. SoundDesigner II also maintains a number of other information resources, but they have no bearing on sampled sound of the type manipulated by Signalyze.
Mono files are in the form of *16-bit signed shorts*. Stereo files provide alternating left and right channel, 16-bit signed shorts. A stereo signal always begins with a left-channel value.

8-bit Formats

SoundEdit Format
Company/Distributor: MacroMind Inc.
File type: FSSD, Creator: SFX!
SoundEdit files maintain a 64-byte resource called 'INFO' (u. c.) and numbered 1000. SoundEdit also maintains a number of other information resources, but they have no bearing on uncompressed sampled sound of the type manipulated by Signalyze.
Sampling rate information is given according to recording frequencies available through MacroMind's A/D converters. The frequencies appear to be 1 = 22.256 kHz, 2 = 11.128 kHz, 3 = 7.418 kHz, 4 = 5.564 kHz, as in SoundWave format**.**
SoundEdit files have separate recording and playback frequency parameters. Both soundEditHdr.relRecFreq and soundEditHdr.relPlaybackFreq are set to Fixed Point 1 when recording and playback speed are normal, >Fixed Point 1 when the speeds are greater and <Fixed Point 1 when they are less. On input, Signalyze calculates a multiplication factor on the basis of the ratio between relPlaybackFreq and relRecFreq, and adjusts the sampling frequency accordingly. On output, Signalyze provides Fixed Point 1 for both of these parameters.
For file length information, Signalyze does not use the soundEditHdr.lftChnnlLength parameter, but instead obtains sound data length directly through the Toolbox command GetEOF(refnum, &SoundLength) to guard against failures from truncated files. On output, current signal length is stored in this parameter.

The compression ratio parameters are ignored by Signalyze.
Mono files are in the form of 8-bit unsigned integers. Stereo files provide alternating left and right channel, 8-bit unsigned integer values.

```
typedef struct soundEditHdr {
    long    res0;               reserved by MacroMind
    long    res1;               reserved by MacroMind
    Fixed   relRecFreq;         relative recording freq., fixed pt format
    Fixed   relPlaybackFreq;    relative playback freq., fixed pt format
    long    compression;        set to 0 by Signalyze in save
    long    mode;               0 for mono, 1 for stereo
    long    windowPos;          set to standard size window by save
    long    sRate;              1: 22kHz, 2: 11kHz, 3: 7kHz, 4: 5kHz
    long    lftChnnlLength;     samples for mono or left stereo channel
    long    rghtChnnlLength;    samples for right stereo channel
    long    res10;              reserved by MacroMind
    long    res11;              reserved by MacroMind
    long    res12;              reserved by MacroMind
    long    res13;              reserved by MacroMind
    long    res14;              reserved by MacroMind
    long    res15;              reserved by MacroMind
} soundEditHdr ;
```

SoundWave Format

Company/Distributor: Impulse Inc.
File type: FSSD, Creator: JOSH
SoundWave files maintain a resource called 'info' (s. c.), numbered 128, according to the data type structure given below.
Sampling frequency is according to recording frequencies available through Impulse A/D converters (1=22.256kHz, 2=11.128kHz, 3=7.418kHz, 4=5.564 kHz).
The speed parameter specifies playback speed which may be faster or slower than recording speed. Just as for SoundEdit format, Signalyze calculates a multiplication factor on the basis of the ratio between this speed and the norm, and adjusts the sampling frequency accordingly on input. On output, Signalyze provides Fixed Point 1 for both of these parameters.
For file length information, Signalyze does not use the soundWaveHdr.waveLen parameter, but instead obtains sound data length directly through the Toolbox command GetEOF(refnum, &SoundLength) to guard against failures from truncated files. On output, current signal length is stored in this parameter.

```
typedef struct soundWaveHdr {
    short   speed;      normal speed 256, faster: >256, slower <256
    short   sRate;      1, 2, 3, or 4
    long    waveLen;    length of sound file
```

```
    Handle    waveH;    handle to where the file read routine puts the signal
} soundWaveHdr;
```

Sound Resource Formats

Company/Distributor: Apple Inc.

Sound resources are stored as 8-bit unsigned sound data in the resource fork of a given file (e.g., a HyperCard stack). Sound resources locate sound data length and sampling rate information in a variable-length header. Signalyze supports sound resource formats capable of encoding simple (one-shot) mono uncompressed sampled 8-bit sounds. This means, a simple format 1 resource, as well as the format 2 resource.

Standard Formats

The following description of the standard header is abstracted from the section on Format 2 'snd ' resources, given on pp. 493-495 of volume V of "**Inside Macintosh"**. Consult "Inside Macintosh" and various Apple technical information sheets for more details.

The sound resource header is in three parts:

Part 1 has the form

```
/* Format 1 */
typedef struct format1SndResHeader {
    short    format;          given as 1 (format 1)
    short    nmbrModifiers;   given as 1 (only one synthesizer used)
    short    resIDfirstMod;   given as 5 (resource ID of
                              sampled sound synthesizer)
    long     initOptfirstMod; given as 1
    short    noSoundCmds;     always 1 in Signalyze
                              (number of sound commands)
} format1SndResHeader;

/* Format 2 */
typedef struct format2SndResHeader {
    short    format;          given as 2 (format 2)
    short    refCount;        ignored
    short    noSoundCmds;     given as 1 in Signalyze
                              (number of sound commands)
} format2SndResHeader;
```

Part 2 is constituted of as many 8-byte sound commands as are specified in initSndResHeader.noSndCmds. This part is ignored by Signalyze.

```
typedef struct midSndResHeader {
    short    bufCmd;          given as 0x8050 (buffer command
```

```
                                    with sampled data)
        short       bufParam;       given as 0
        long        offset;         14 (offset to end of header, start of data)
    } midSndResHeader;
```

Part 3 has the form

```
    typedef struct endSndResHeader {
        long        dataOffset;     0
        long        dataLength;     signal length
        long        sRate;          Fixed point sampling rate in Hz
        long        loopStart;      given as signal length - 2
        long        loopEnd;        given as signal length - 1
        char        standard[1];    given as 0 (indicates standard
                                    sample encoding)
        char        baseNote;       given as 0x3C (corresponds
                                    to middle C in synthesizer)
    } endSndResHeader;
```

Sound data follow the endSndHeader with the offset given in endSndResHeader.dataOffset (the offset is typically 0). Sound data is in **unsigned char**s, where the ideal resting level is 128, the most negative amplitude is 0 and the most positive amplitude is 255.

Type "B" or "generic" Sound Resource Format

There are still some sound resource files around that do not conform to the Apple Standard laid down in 1988 and published in Volume V of **Inside Macintosh**. To be able to read these files, a generic sound resource format was created

The header is assumed to be 42 bytes long (which is appropriate to many, but not all such files) and a default sampling rate of 22256 Hz is furnished. The user is expected to set the appropriate sampling rate through experimentation, and by varying the sampling rate given in Signal>Signal Information or obtained through SHIFT-click on the sampling rate indicator.

Import / Export (Numeric, ASCII, TEXT) Format

Company/Distributor: Public

File type: TEXT, Creator: *[unspecified]*

This format has many names, but it has essentially just one form. It is a series of numbers, written in normal text characters, and separated by a specific character. Many signal analysis, spreadsheet, mathematics,

---

[1] Notice the char type. This is according to Apple's Oct. 1988 update and correction to the Sound Manager chapter in Inside Macintosh V which lists an *int* instead of the two *chars* in the final part of the sound resource header. Neither Think C 3.0 nor 4.0 has made this correction in its headers, so you may have to correct the header information.

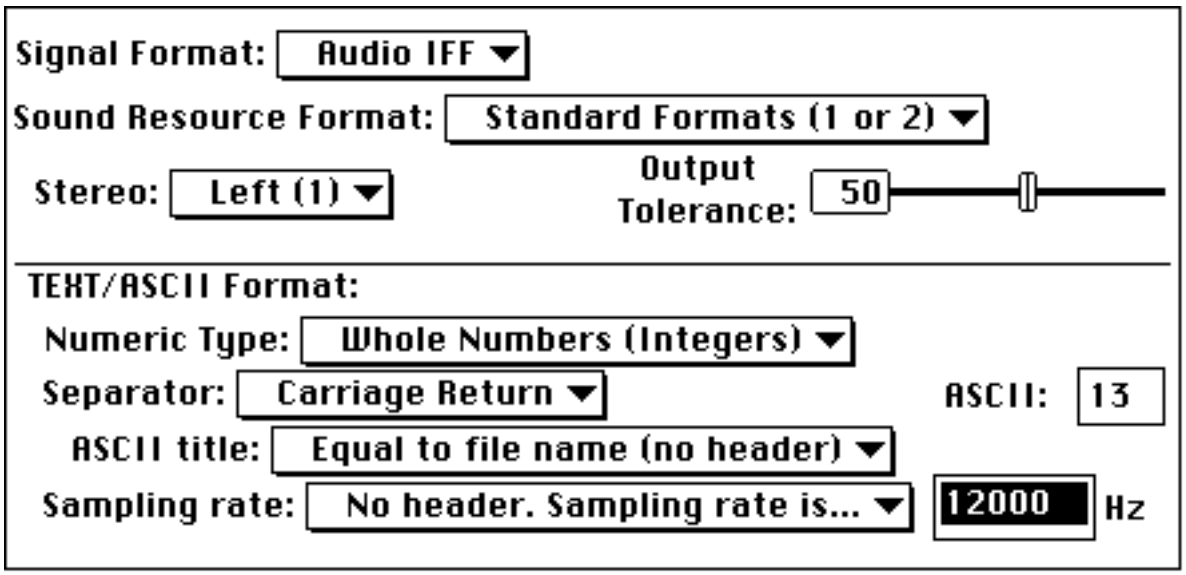
graphics and text processing programs on all types of computers can read and write this format.

The Headerless Format (Dialog option or File name option)

At its simplest, signals in this format are just a list of numbers. If the separating character is a RETURN (shown here as "¶"), the numbers appear on succeeding lines:

23789¶
23654¶
23549¶
etc.

Files of this type can be entered into Signalyze with the options as shown below in File>File Setup. The name of the signal will be taken from the file name and the sampling frequency will be taken from the edit field (showing 12000 Hz here). Of course, the correct sampling frequency must be specified by the user.



The Header Format (File header option)

To simplify the entry of file name and sampling frequency, a second format is defined by Signalyze. In this format, the list of numbers is preceded by a header containing the signal name and the sampling rate. A signal called "long-a" with a sampling frequency of 10 kHz might look like this:

long-a¶
10000¶
23789¶
23654¶

23549¶
etc.

Files of this type can be entered into Signalyze with the following options in File>File Setup:

ASCII title: In file header
Sampling rate: In file header.

Specifications for Import/Export Format

Signal name: Maximally 30 characters long, any string of ASCII characters

Whole/Decimal: Numbers can be whole or decimal numbers. If decimal, the decimal portion of the number is disregarded on input and set to zero on output.

Numerics: Legitimate signal values range from -32,766 to +32,766. The number +32,767 is interpreted to mean "missing value", i.e., a sample that is there, but is not shown.

Sampling rate: Coded according to the same convention as Signalyze files (see above).

Separators: Any single ASCII character, other than: numerics, minus sign, period.

Space character: *(New in Version 2.0):* When one or more space characters ("blank", ASCII 32) surround a number, they are discarded, provided the space character does not serve as separator. If the space character is specified as separating character, numbers cannot be surrounded by blanks.

End of file: In accordance with the Macintosh standard, a last separator is not required. *A zero is inserted as the last sample if the file ends with a separator*.

FileConverter Signal Formats

The additional signal formats supported by the "FileConverter" program are documented in a file found in the "FileConverter" folder on the Signalyze distribution diskette. The current update of this file is available on the Signalyze electronic servers (see "Updates" in the section "Installation Suggestions").

Label File Format

Here is a full description of the Signalyze label file structure. This should let you convert labels written in other environments to Signalyze format, and should let you perform automated extractions of information on the basis of Signalyze labels. The Signalyze format for labels is hereby placed into the public domain. Its official name is “Signalyze Label Format”.

Labels are stored in the same folder as their corresponding signal files. Label files have the same name as their corresponding signal files, plus the extension “.lbl”. The body of the signal name must not exceed 31 characters. As usual in the Macintosh environment, the signal name can include blanks. The corresponding signal files can be of any format supported by the Signalyze application, i.e., Signalyze, AIFF, AudioMedia, etc.

The number of labels you can create is very large. Since the numeric limit on unsigned longs is 4,294,967,296, and since each can designate another millisecond in the signal, this is the number of possible labels. Also, Signalyze labels can theoretically designate points in the signal at up to 4.2 million seconds from the start, i.e., over 1193 hours of continuous recorded material. Internally, each label consumes 32 bytes, and on disk, the space consumed depends on the character length of the various text strings making up the label description.

Label files are TAB-delimited text files, and can thus be read with any text editor, including the one incorporated in Signalyze. In label files, numbers are given in their ASCII form. In the following structure definitions, ASCII numbers are characterized in terms of the numeric types they assume internally, and hyphen-hooks (¬) take the place of Tab’s. “Segment labels” are labels that designate a signal segment, in contrast to “position labels” which designate a particular position within the signal.

Line 1:

```
char    signalName[31]¬  /* signal name -- for identification only (string) */
long    numLabels¬       /* number of labels */
short   txFont¬
short   txSize¬          /* text font and text size (Apple’s coding) */
short   levelColor1¬     /* level colors (1-7). 1: yellow; 2: red; 3: magenta;
                                4: green; 5: cyan; 6: blue; default: black */
char    levelName1[14]¬  /* level names (strings) */
short   levelColor2¬     /* level colors and names for subsequent levels */
char    levelName2[14]¬
short   levelColor3¬
char    levelName3[14]¬
short   levelColor4¬
char    levelName4[14]¬
short   levelColor5¬
char    levelName5[14]¬
```

```
short       levelColor6¬
char        levelName6[14]¬
short       levelColor7¬
char        levelName7[14]¬
short       levelColor8¬
char        levelName8[14]¬
short       levelColor9¬
char        levelName9[14];
```

Succeeding lines (one line per label)

```
char          labelText[30]¬    /* the label text */
short         level¬            /* the label's level (1-9) */
unsigned long timeStamp¬        /* the label's center (ms, integer) */
unsigned long beginLabel¬       /* for segment labels: onset; else: timeStamp */
unsigned long endLabel¬         /* for segment labels: offset; else: timeStamp */
short         top;              /* the label's vertical position (out of 64 pixels */
```

Numeric Formats

Via Think C numeric routines, Signalyze performs all floating point operations in **double**, which is the most efficient floating point calculation mode on the Macintosh. In a few time-critical routines (e.g., IIR filtering), external routines are called that use coprocessor-specific floating point math. Here are the various numeric data types, as used in the different environments:

Think C	Size	APPLE	Use in Signalyze
char	8 bits	CHAR	sound reproduction, 8-bit formats
short (or **int** or **short int**)	16 bits	INT or WORD	signals, fast calculations
long (or **long int**)	32 bits	LONGINT	memory addresses, fast calculations
float	32 bits	SINGLE	not used
short double	64 bits	DOUBLE	not used
double	96	EXTENDED DOUBLE	floating point calculations

Time Calculations

All Signalyze time calculations are performed via the *microsecond standard* by means of floating point operations. E.g., to determine the time of a given sample, the microsecond is determined on the basis of the sample number, the sampling frequency and the zoom setting. Or again, to determine the time corresponding to the position of the cursor, a rather complex calculation is performed involving the number of pixels between the cursor line and the left edge of the signal, the offset of the left edge with respect to the beginning of the signal, the zoom setting and

the sampling rate. The microsecond standard was chosen to minimize rounding errors within such complex chains of calculations.

The Signalyze Text Editors

When the *Text Window* is in front, users have access to a complete small text editor that implements most of the Macintosh integrated text features (editing features, like cut and paste, font and font size manipulations, window scrolling, etc.). Text can be saved as TEXT files (i.e., files with the file type 'TEXT' and a user-specifiable creator signature, e.g., 'MSWD').

Signalyze can open and read all files with the file type 'TEXT', as long as the file is shorter than 32k and the number of lines does not exceed 16,000. Similarly, generated text in the Text Window cannot exceed 32,000 characters. Text is stored in heap memory (memory reserved for applications), and thus consumes calculation memory available to Signalyze's signal analysis routines. This, together with system limits on the Macintosh integrated text manipulation routines, dictate the 32 k size limits.

Text in the Text Window display can be up to 600 pixels wide before it is word-wrapped. This permits easy horizontal arrangement of multiple TAB-separated columns. However, on smaller screens, this means that text can disappear to the right and that you must work with the horizontal scroll bar to make the right portion of the text visible. (The user can set the word wrap limit through Edit>Preferences.)

Cut, copy and paste operations involve a text buffer (Apple terminology: "scrap") that is set to a maximum of 10,000 characters. Larger text transfer operations must be effected with a word processor specifically designed for this type of task. Text material can be copied to and from other programs via the usual copy and paste operations.

Why does the Signalyze text editor let you open only one text document at a time? To save memory for signal operations, of course—and to prevent confusion. Since the open text document can be a target for the saving of statistical and numeric evaluations, it is much easier for both the program and the user to have only one text document open at any time. Further text documents can easily be opened with another text editor operating in the background.

In the *Signal Window*, Signalyze maintains a less sophisticated text editor offering the Macintosh editing features, but no file access. Window scrolling is performed with the up and down arrows on the keyboard. Theoretically, text in this window can grow to 32 k, but only a few lines are visible at any one time.

The Most Common Questions about Signalyze

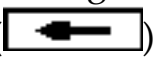
(or "All you ever wanted to know about Signalyze and didn't dare ask")

On Usage:

- **I don't read manuals and I think programs should be self-explanatory. Signalyze is not at all "obvious" to use. Why?**

Doing signal analysis is much like driving a car. It does become "obvious" after awhile, but till you get there, you can make a lot of very costly errors. The manual is there to keep you from making most of these errors. It's up to you to profit from its wealth of information. But to help you along, there is also a rather complete on-board Help. Even if you refuse to read manuals, maybe you can stop every once in awhile to check out the Help.

- **When I open a second signal, the first one disappears. Why?**

If you specify "New Card" in Signal>Display Setup, the second signal is shown on a second *card*. You can get the first signal again by pressing on the left Card Switch button ().

- **After deleting a signal, there is an empty slot. When I then open a new file, the file does not show up in the empty slot, but is placed at the end of all the other signals.**

Remember that the Destination Slot Button ( ) has highest priority for signal placement. If you want the newly loaded signal in the now empty track, select the track with the Destination Slot Button or by clicking in the empty track. If the active Destination Slot Button is next to an unoccupied track, the new signal is placed there.

- **I understand the program to be "memory-greedy", and since I will want to use it to extract intonation contours over fairly long strings (10-15 syllables), I wonder if you have suggestions as to how to minimize the amount of memory needed per string.**

"Memory-greedy" may be one way to characterize Signalyze memory use, but in fact, substantial memory demands are the price to pay for a fast-response interactive program that stores nearly everything in RAM. I could have conceived Signalyze around disk swaps, but I decided not to for a number of reasons. For one, that would have slowed its operation

down considerably, for another, I don't think that's the way to go in a world of greatly lessened memory chip cost. Finally, I don't think it's necessary in the Macintosh world where we have virtual memory as a standard feature for Macs with a PMMU, or utility programs like "RAM Doubler".

So my suggestion is this. Consider putting your Macs up to *4, 5 or 8 Mb*. That should handle signals with 10-15 syllables just fine. If you still run out of memory, please consider *downsampling* at some suitable frequency. Signalyze near-instantaneous downsampling to nearly any target frequency. Or switch to *virtual memory*. It does slow Signalyze down a bit, but it does the job.

With these solutions you have such a panoply of possibilities that under most circumstances, you should be able to find a suitable compromise between Signalyze memory requirements, your signal analysis demands and available resources.

- **I have a Mac II ci (with cache card) running off of the internal video. Would a separate video card or an accelerated video card hasten the output of spectrographs? I use all the techniques you suggest to get faster output, but there are times when I need to monkey with 256 shades in high resolution mode.**
- To give you the brutal answer right away: Upgrade to the strongest and fastest machine you can afford. The slowdown is not exclusively due to the video, but to general computation tasks. The spectrogram routine is a very complex routine, given all the options that are available in Signalyze. Video processing and the FFT algorithm take a certain proportion of the total computational time, and much further time is used up with sorting out which option applies and which doesn't, where to take the time window, how to weigh it, and how to interpret the FFT result. Also, some color interpolation has to be calculated for large zooms. The routine has been optimized about as much as possible, for example there is very little calculation in floating point. I've nearly exhausted what I can do on this end to speed things up, the only thing left to do is to add brute power.

- **I use the AudioMedia II board to digitize my sounds. Typically I use a sampling rate between 27777 and 44100. I've noticed that the information bar always gives me the correct sampling rate, but it tells me that I'm using 12-bit format when I sample at 27777, and 15-bit format when I sample at 44100. The AudioMedia people swear that all signals are digitized in 16-bit format, regardless of the sampling rate. Any ideas about why this is happening?**
- Signalyze usually establishes bit width *empirically*. That is, even though you may be recording a 16-bit signal, your signal amplitude may

so low that it only takes 14 or 12 bits to encode all the gradations contained in it. You'll find that the empirically derived bit width increases with additional volume. It attains 16 bits if your signal shows clipping.

Bit width is derived empirically in Signalyze to be able to control for varying signal strengths in audio output, spectrograms and other spectral analyses, to guard against math overflow in a large variety of routines, and to render Signalyze compatible with as many input devices and file formats as possible. However, if you really want 16 bits all the time, you can request it by setting "Force 16 bits" in File>File Setup.

And while we're on standardization, here's a common answer to another common question:

- **Why does the bit width (or the offset, or the sampling frequency) change when I save a signal in Sound Designer II format?**

- Please remember that there are only three formats in Signalyze that impose no major constraints on bit width, offset and sampling frequency:

— Signalyze format
— AIFF format
— ASCII format

All other formats must be saved at specific sampling frequencies. In addition, some impose a bit width of less than 16 bits or a specific offset. To accommodate such formats, Signalyze thus performs an automatic standardization where signals are bit shifted to the correct bit width and offset, and up- or down-sampled to the nearest acceptable sampling frequency.

Consequences: you have reduced control over the changes you may have made in your signal. Also, you necessarily degrade the signal through the resampling process. So—save your intermediate versions of the files either in Signalyze or AIFF format and convert to a standardized format only at the very end of your processing chain.

- **Why does it take so long to "Quit" the program on my MacPlus/Classic/SE?**

At the end of the program, the configurations specified in the setup dialogs are saved to disk and stored in the application file. Especially on floppy disk systems, this can take a few seconds.

- **How can I be sure that the distribution diskette is free of viruses?**

Considerable security precautions prevail at InfoSignal to ensure a virus-free environment on the hard disk on which Signalyze is being developed. Public domain software is tried out on other disks before it is used, the most up-to-date versions of anti-virus programs are in use, and each final distribution version of Signalyze is visually examined for infection with a resource editor prior to shipping. All reasonable

precautions are being taken, but as everyone knows, there can be no guarantees.

On Features:

• **Why is there no "Undo" command?**
Although the "Undo" command is clearly one of the signs of user friendliness (some would say: "One of the great signs of civilization"), it also involves an "undue" ('xcuse the pun) programming effort, and in a signal analysis environment, it would gobble massive amounts of memory. Storing everything automatically to disk would hopelessly slow down the smaller types of Macs. So far, other aspects of the program have had greater priority. In a future version, when memory is even more plentiful than now, some form of "undo" will probably make more sense than right now.

• **Why can't I get frequency or amplitude values from clicking in large spectrograms?**
It really must be emphasized that ***for research purposes, one should never rely on read-outs obtained by clicking in spectrograms***. The danger of working with seriously erroneous values is great. Spectrograms are generated either from (rather imprecise) 256-point or 512-point FFTs and many features are highly interpolated (the greater the vertical zoom and the wider the band, the greater the interpolation). Spectrograms are here primarily for quick-and-dirty looks at the overall features (in the case of spectrograms on the Signal Window) or for publication or documentation purposes (in the case of spectrograms in separate windows). If one were to pick peaks or features off the spectrograms in any systematic fashion, *substantial measurement error would be just about guaranteed*.

In order to obtain values for research purposes, the only reliable way is to generate spectra (spectral sections) at the desired places. Pertinent frequency and amplitude values can then be generated from the sections by clicking. Spectra are generated with 1024-point FFTs and thus offer a *reasonably reliable* (i.e., about as best as can be got by digital means) estimate of frequency peaks and amplitude values.

• **When I generate a 1000-Hz tone with SoundEdit, Signalyze's spectral analysis routine pretends it is a 1012-Hz tone**
As well it should. You probably generated the tone at 22 kHz, which means that each cycle measures exactly 22 samples. However, SoundEdit's (and the Mac's) sampling rate is 22,256 samples/second, so that tone *actually* begins a new cycle every 9.88 ms. Translated into frequency measures, that makes a 1011.63 Hz-tone. Consider also that the

FFT has some inherent imprecision, so the actual FFT-measured frequency will be a few Hz off.

- **I get duration measures with SoundEdit that are very different from those of Signalyze**

So do I (at least with SoundEdit 2.0.3), and particularly so at very large zoom settings. But I can only answer for Signalyze's measurements. You can verify the following measures yourself. Generate a 1000-Hz tone with SoundEdit's Tone Generator option. Take the signal into Signalyze. Each cycle will be 22 samples or 9.88 ms long. Measure the times at the cursor and by blackening. You'll probably find that the measured times are exactly right.

- **What speed of hard disk I needed for the AudioMedia board? Do you by any chance know if the IIci or fx have a fast enough disc?**
- Any hard disk that comes with a recent Macintosh series II has plenty of leeway in its access times to store sound continuously via the AudioMedia software. Signalyze can only record into RAM. Many users with heavy recording schedules do all the recording with AudioMedia and then analyze with Signalyze. The file formats are compatible between programs.

License Agreement

This legal document is an agreement between you, the end user, and InfoSignal Incorporated. The agreement includes the software license, disclaimer of warranty and limited guarantee and warranty, and an acknowledgment (collectively the "Agreement"). If you do not agree to the terms of this agreement, promptly return the complete software package (disk(s) and manual in unused condition) to InfoSignal Inc., at the address where you obtained it.

In order to preserve and protect its rights under applicable law, InfoSignal Inc. does not sell any rights in software. Rather, InfoSignal Inc. grants the right to use software by means of a software license. InfoSignal Inc. specifically retains title to all the software it produces, texts of the accompanying written materials, as well as to the trademarks "Signalyze™", "InfoSignal™", and the InfoSignal brain-signal icon.

SOFTWARE LICENSE

1. Grant and type of license.

a. Single-user ("individual") license : In consideration of payment of initial license and license upgrade fees (which may be waived in certain cases), InfoSignal Inc., grants to you, the licensee, a non-exclusive right to

put into service this copy of a software program or software program library (hereinafter the "software") *on a single computer at a single location*.

b. Departmental and organizational licenses. In consideration of payment of initial license and license upgrade fees (which may be waived in certain cases), InfoSignal Inc., grants to the members of the licensed departmental or organizational group a non-exclusive right to put into service any number of copies of a software program or software program library (hereinafter the "software") on any computer installation owned by the organization in a single, geographically coherent site. A *department* is defined as a separate budget entity (department, laboratory, research group, etc.) within a larger organization. An *organization* is defined as a geographically coherent entity (e.g., a university, a corporation). Geographically separate entities of the same overall organization (e.g., different campuses of the same network of state universities) are counted as separate organizations. However, off-campus laboratories or field stations (up to a maximum of one per department and three per organization) are deemed to be part of the organization on which they depend directly for their financial support.

"Putting into service" means installing the software in such a manner that its code is in the process of execution. Installing the software on a server solely for distribution purposes does not constitute "putting the program into service". You are responsible for insuring that the number of concurrent users does not exceed the numbers licensed. InfoSignal Inc. reserves all rights not expressly granted to Licensee.

2. Ownership of software. The Licensee owns the magnetic or other physical media on which the software is originally or subsequently recorded or fixed, but InfoSignal Inc., retains title and ownership of the software recorded on the original disk copy(ies) and all subsequent copies of the software, regardless of the form or media in or on which the original and other copies may exist. This license is not a sale of the original software or any copy.

3. Copy restrictions. This software and the accompanying written materials are copyrighted under international copyright. Unauthorized copying or selling of the software, including software which has been modified, merged, or included with other software programs, or of the accompanying written materials, is expressly forbidden. You may be held legally responsible for any copyright infringement which is caused or encouraged by your failure to abide by the terms of this License. You may make as many backup copies as is deemed necessary for proper use of the materials, appropriate to your type of license.

4. Use restrictions. As the Licensee, you may physically or electronically transfer the software from one computer to another, within the extent permitted by the type of license (single-user, departmental, organizational). You may not distribute copies of the software or

accompanying written materials to others. You may not modify, adapt, translate, reverse engineer, decompile, disassemble, or create derivative works based on the software. You may not modify, adapt, translate or create derivative works based on the written materials without the prior written consent of InfoSignal Inc.

5. Transfer restrictions. This software is licensed only to the Licensee, and may not be transferred to anyone without the prior written consent of InfoSignal Inc. Any authorized transferee of the software shall be bound by the terms and conditions of this License and Limited Warranty. In no event may you transfer, assign, rent, lease, sell or otherwise dispose of the software on a temporary or permanent basis except as expressly provided herein.

6. Termination. This License is effective until terminated. This License will terminate automatically without notice from InfoSignal Inc. if you fail to comply with any provision of the License.

7. Update policy. InfoSignal Inc. may create, from time to time, updated versions of the software. At its option, InfoSignal Inc. will make such updates available to Licensee and transferees who have paid the applicable license upgrade fee(s).

8. Miscellaneous. This License is governed by the laws of the Canton of Vaud, Switzerland, and shall benefit InfoSignal Inc., his successors and assigns.

DISCLAIMER OF WARRANTY AND LIMITED GUARANTEE AND WARRANTY

The software and accompanying written materials (including instructions for use) are provided “as is” without warranty of any kind. Subject to the exceptions listed below, InfoSignal Inc. does not warrant, guarantee or make any representations regarding the use, or the results of use, of the software or written materials in terms of correctness, accuracy, reliability, currentness, or otherwise. The entire risk as to the results, performance and potential side effects of the software is assumed by you.

Limited Guarantee: If the software is found to be incapable of performing the operations substantially as detailed in the written descriptions accompanying the software, and if this defect cannot be eliminated within one month of receipt of a detailed and replicable description of the defect, you may elect to request a full refund of the currently applicable license fee. This guarantee specifically excludes sound input- and output routines using non-standard equipment (equipment not produced by Apple Inc.).

The applicable license fee is the payment made since the software’s last major upgrade. Refund is granted upon remission of the following documents: A detailed description of the problem, file(s) on diskettes posing problems and the original current distribution disk(s). Your own

diskettes will be returned. Postage for the sending of these materials to InfoSignal Inc. must be assumed by you.

Limited warranty: InfoSignal Inc. warrants to the original Licensee that the disk(s) on which the software is recorded is/are free from defects in materials and workmanship under normal use and service for a period of ninety (90) days from the date of delivery as evidenced by a copy of the receipt. Further, InfoSignal Inc. hereby limits the duration of any implied warranty(ies) on the disk to the period stated above. Some states do not allow limitations on duration of an implied warranty, so the above limitation may not apply to you. If failure of the disk has resulted from accident, abuse, or misapplication, InfoSignal Inc. shall have no responsibility to replace the disk. In the event of replacement of the disk, the replacement disk will be warranted for the remainder of the original warranty period or thirty (30) days, whichever is the longer.

The above are the only guarantees and warranties of any kind, either express or implied, including but not limited to the implied purpose, that are made by InfoSignal Inc. on this product. No oral or written information or advice given by InfoSignal Inc., its dealers, distributors, or agents shall create a guarantee or warranty, or in any way increase the scope of this guarantee and warranty, and you may not rely on any such information or advice.

Neither InfoSignal Inc., nor anyone else who has been involved in the creation, production or delivery of this product shall be liable for any direct, indirect, consequential or incidental damages (including damages for loss of business profits, business interruption, loss of information, and the like) arising out of the use or inability to use such product, even if InfoSignal Inc. has been advised of the possibility of such damages.

In the event of any conflict between any provision in this limited guarantee and warranty and any applicable provincial or state legislation, applicable legislation takes precedence over the contravening provision. This Limited Guarantee and Warranty is governed by the laws of the Canton of Vaud, Switzerland, and shall benefit InfoSignal Inc., his successors and assigns.

ACKNOWLEDGMENT

You acknowledge that you have read this AGREEMENT, understand it, and agree to be bound by its terms and conditions. You also agree that license and disclaimer of warranty and limited guarantee and warranty is the complete and exclusive statement of agreement between the parties and supersede all proposals or prior agreements oral or written, and any other communications between the parties relating to the subject matter of the AGREEMENT.

InfoSignal™ Inc., C.P. 73, CH-1015 LAUSANNE, Switzerland

References

Abberton, E., & Fourcin, A. (1984). Electrolaryngography. In C. Code and M. Ball (Eds.), *Experimental Clinical Phonetics*.(pp. 62-78). Croom-Helm.

Apple Computer, Inc. (1987). *Human Interface Guideline: The Apple Desktop Interface.* Addison-Wesley.

Apple Computer, Inc. (1988). *Inside Macintosh™: Volume V*. Addison-Wesley.

Hess, Wolfgang (1983). *Pitch determination of Speech Signals (Algorithms and Devices).* Springer-Verlag.

Pohl, I., & Edelson, D. (1988). A to Z: C language shortcomings. *Computer Languages, 13,* 51-44.

Schildt, Herbert (1987). *C: The Complete Reference.* Berkeley, CA: Osborne McGraw-Hill.

Appendix

Summary of Extension Codes for New Signals

When the program creates a second signal from a first signal, it appends an extension to the first signal's name. E.g., a spectrum produced from the signal "abc" is named "abc.spctrm".

avgSpctrm	long-term averaged spectrum
ct-l	complex transformation, limit
ct-r	complex transformation, resample
ct-v	complex transformation, derivative
ev-m	envelope, power (mean square)
ev-p	envelope, peak-to-peak
ev-r	envelope, RMS (root mean square)
fltr-Hamm	filter, Hamming 7-point fixed
fltr-IIR	filter, Butterworth or Chebychev
fltr-mvavg	filter, moving average
fltr-trngl	filter, Triangular 3-point fixed
Fo-AC	fundamental frequency, autocorrelation
Fo-FFT-cmb	fundamental frequency, FFT-comb
Fo-TSA	fundamental frequency, temporal structure analysis
pst	pasted signal
spctrm	spectrum
spl-avg.d0	spline, average, derivative 0 (signal)
spl-avg.d1	spline, average, derivative 1 (velocity)

spl-avg.d2	spline, average, derivative 2 (acceleration)
spl-tns.d0	spline, tension, derivative 0 (signal)
spl-tns.d1	spline, tension, derivative 1 (velocity)
spl-tns.d2	spline, tension, derivative 2 (acceleration)
st-a	simple transformation, add
st-c	simple transformation, cos
st-d	simple transformation, divide
st-e	simple transformation, exchange sign
st-f	simple transformation, shift left
st-g	simple transformation, shift right
st-i	simple transformation, sin
st-l	simple transformation, log 10
st-m	simple transformation, multiply
st-n	simple transformation, natural log
st-p	simple transformation, power
st-q	simple transformation, sqrt
st-r	simple transformation, reciprocal
st-s	simple transformation, subtract
st-t	simple transformation, tan
zc	zero crossing

Summary of Keyclick Combinations

PRINCIPLES OF KEY ASSIGNMENTS

SHIFT
Gives popup menus, edit popouts, and dialog boxes

OPTION/COMMAND
Gives binary options, e.g., narrow vs. wide band spectrograms

COMBINATIONS OF KEYS
Shows values in Edit Window

KEY PRESSES AT STARTUP

• Hold OPTION-COMMAND: Put Signalyze into read-only mode (the chosen options are not saved).

CLICKS IN SIGNAL DISPLAY

• Simple click: Line cursor placement
• SHIFT-click: Extend signal selection
• SHIFT-COMMAND-click: Add time indication to Edit Window
• SHIFT-OPTION-COMMAND-click: Show bottom number of Signal Control Panel in Edit Window
• SHIFT-OPTION-click: Erase Edit Window
• Double-click: Transfer Edit Window to Text Window, add Return
• OPTION-click: Zoom left to cursor
• COMMAND-click: Zoom right to cursor

• OPTION-COMMAND-click: Displace left edge of display to cursor position

CLICKS IN MINI-DISPLAY

• OPTION-click: Zoom left to cursor
• COMMAND-click: Zoom right to cursor
• OPTION-COMMAND-click: Displace left edge of display to cursor position

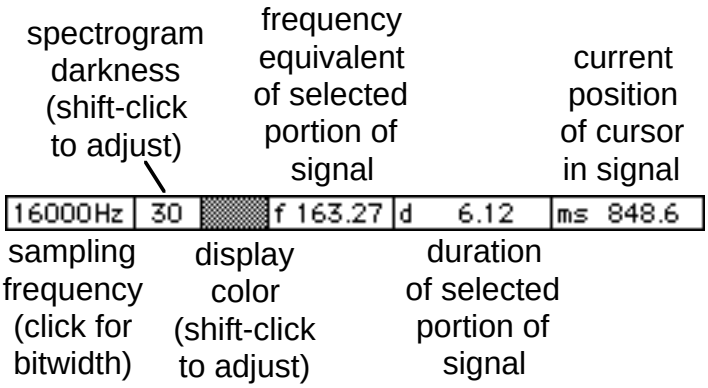
CLICKS IN SPECTROGRAM/CEPSTROGRAM

• Simple click: Line cursor placement
• SHIFT-click: Extend signal selection
• SHIFT-COMMAND-click (Spectrogram): Show time and frequency in Edit Window
• SHIFT-COMMAND-click (Cepstrogram): Show time, f0 cycle length, and frequency in Edit Window
• SHIFT-OPTION-click: Erase Edit Window
• Double-click: Transfer Edit Window to Text Window, add Return

CLICKS IN SPECTRUM/CEPSTRUM

• Simple click: Line cursor placement
• SHIFT-click (Spectrum or Cepstrum): Show frequency and time where spectrum taken in Edit Window
• SHIFT-OPTION-COMMAND-click (Spectrum): Show dB in Edit Window
• SHIFT-OPTION-COMMAND-click (Cepstrum): Show Fo cycle duration in Edit Window
• SHIFT-OPTION-click: Erase Edit Window
• Double-click: Transfer Edit Window to Text Window, add Return

GENERAL CONTROL PANEL INDICATORS



— Frequency/Bitwidth Indicator (leftmost)
• No key: Toggle between current signal's sampling frequency and bitwidth
• SHIFT: Edit sampling frequency. This manipulation changes only the overall pitch and does not change the signal itself.

— Spectrogram Darkness Indicator (2nd)
• SHIFT: Set spectrogram darkness

— Color Indicator (3rd from left)
(only operational on Macs with color/grayscale)
• SHIFT: Set color of selected plot

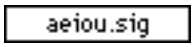
— Frequency Indicator ("f", 4th from left)
• No key: Show frequency equivalent of duration of signal selection in Edit Window.
Only operational if duration > 0.
(Frequency equivalent = 1000 / duration in ms).

— Duration Indicator ("d", 5th from left)
• No key: Show duration of signal selection in Edit Window

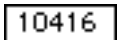
— Time Indicator ("s/ms/μ", 6th from left)
• No key: Show time at the cursor in Edit Window

— Card Indicator
• No key: Jump to card x

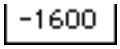
SIGNAL CONTROL PANEL INDICATORS



Signal Name (top row)
• No key: Show name in Edit Window
• SHIFT: Edit signal name



Top Number (second row)
• No key: Show top number in Edit Window



Bottom Number (third row)
• No key: Show bottom number in Edit Window
• SHIFT: Edit bottom number (regular signals only)

GENERAL CONTROL PANEL BUTTONS

TOP ROW:

Record Menu Button (leftmost)
• No key: Store new menu in programmable button

to

Play Back Menu Buttons (others)
• No key: Call stored menu
• OPTION: Show currently-stored menu (Top number: Menu, counted from the left.
Bottom number: Menu item, counted from the top. Two zeroes: "No menu stored")

All top row buttons: You can associate a specific letter of the extended ASCII alphabet with any of these buttons through Edit>Preferences.

SECOND ROW:

Pitch Extraction Button (left)
• No key: Extract pitch from current signal

• SHIFT: Pitch Extraction Setup

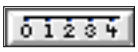
Spectrogram Button (middle)
• No key: Make spectrogram
• SHIFT: Spectral Analysis Setup
• OPTION: Wide band spectrogram
• COMMAND: Narrow band spectrogram

Spectrum Button (right)
• No key: Make spectrum
• SHIFT: Spectral Analysis Setup

THIRD ROW

/ / Edit Button (left)
• No key: Three-way toggle between
- editing text in Edit Window
- editing current signal
- one-shot text transfer mode
(transfers entire contents of Edit Window
to Clipboard, stays in mode till toggled)

Abort Button (2nd from left)
• No key: Aborts a current user action or an ongoing process, erases edit window

 Scales Button (3rd from left)
• No key: Toggles scales on and off
• SHIFT: Scales dialog box
• OPTION: Toggle scales for y-axis
• COMMAND: Toggle scales for x-axis
• SHIFT-OPTION: Toggle gridlines for y-axis
• SHIFT-COMMAND: Toggle gridlines for x-axis

Left and Right Card Buttons (extreme right)
• No key: Turn “cards” left or right
(The number in the middle shows which
card you are on)

SIGNAL CONTROL PANEL BUTTONS

Signal Selection Button (upper left)
• No key: Get signal from other track
• SHIFT: Load signal via Open File dialog
• OPTION: Load signal via Open Sound Resource dialog

Display Size Button (upper middle)
• No key: Toggle between tall and regular signal
• SHIFT: Control vertical zoom

Horizontal Line Button (upper right)
• No key: Turn horizontal cursor on and off
• OPTION: Horizontal cursor follows click
• COMMAND: Horizontal cursor stays stationary

Playback Button (lower left)
• No key: Play sound
• SHIFT: Set playback volume level
• OPTION: Play whole signal
• COMMAND: Play selected segment
• OPTION-COMMAND: Play signal except for selected segment

Signal Plot Button (lower middle)
• No key: Replot
• OPTION: Replot as connected line plot
• COMMAND: Replot as minmax plot
• SHIFT-COMMAND: Replot whole signal at display width
• OPTION-COMMAND: Replot selected segment at display width
• SHIFT OPTION: Toggle between normal/narrow plot (narrow plots are useful for showing pitch extractions)

Delete Signal/Display (lower right)
• No key: Delete selected SIGNAL
• COMMAND: Delete selected DISPLAY (i.e., selected COPY of the signal display)

Index

% valid observations 50, 143
'snd ' resources 125
+ / - 190
3-point triangular filter 200
3d spectrogram through NIH image 96
7-point hamming filter 200
8- and 16-bit signal formats 90
8-bit files 123
8-bit format 90, 129, 219
8-bit signals 72
12-bit format 91
12-bit signals 72
12 / 16-bit format 129, 217
16-bit Apple Sound Manager 10
16-bit format 91, 229
16-bit signals 72
16-to-8 bit recoding 126

Abort button 33, 115
About Signalyze 120, 215
accelerate or slow down speech 33
accelerated video 229
acceleration 196
accelerator boards 67
active destination button 133
add 189
adding noise to a signal 76
additions 7
Afficher graphique 184
Afficher les étiquettes 151
Afficher statistiques 143
Afficher valeurs 140
Aide 121
align or time-offset signals 73
aligning columns 45
Alle ausser dem ersten löschen 161
Alle signale löschen 161
Alles auswählen 134
Als Tonressource speichern 125
alternate pixels 154
amplifier 13
amplitude 140
amplitude differences 177
amplitude envelopes 78
amplitude measurement 76
Amplitudenhülle 195
analog vs. digital filters 83
analysis window 179
analysis window width (why control) 180
animal sounds 19, 171
anti-aliasing 57, 171, 176
anti-aliasing filtering 83
Apple input devices 208
Apple Macintosh interface guidelines 210
Apple menus 120
Apple sound input device 12, 34
arithmetic and transcendental transformations 76
arrow keys 109
articulatory movements 18
ASCII 223
ASCII-format signal with header 95
ASCII-format signal without header 96
ASCII-format signals 94
Audio Interchange File Format (AIFF) 217
audio setup 158
AudioMedia 35
Ausschneiden 132
auto-repeat 32
autocorrelation 54, 56, 65, 165
AutoEdit 203
AutoEdit setup 202
automatic gain control (AGC) 209
automatic trimming 75
automatic window sizing 148
Automatische Bearbeitung 202, 203
AV-machines 10
average spline 81, 82, 197
averaged spectrum 175

Background priority 145
backup media 105
batch load 103
Beenden 132
Bei Programmwechsel verbergen 185
bias 77, 192
Bild öffnen 126

Bild speichern 126
bit mapping 213
bit width 77, 230, 192
black & white 174
black-and-white spectrograms 67
blanks 45, 128, 139
blitting 213
buffer 132
built-in synthesizers 158
Butterworth 86, 200
buttons 113

Cacher en multitâche 185
Cacher fenêtre de signal 184
Cacher fenêtre de texte 185
Cacher toutes les fenêtres 185
calculation memory 102, 147, 215
card 70
card indicator 112
card selection 39
card switch buttons 116
cards 212
cautionary dialogs 162
cepstrum 54, 55, 174, 181
change a specific signal value 44
change default 155
change other plots 155
change selected plot 155
changing speed 202
changing title, amplitudes and sampling frequency 72
Chebychev 200
Chooser 99
clear 133
clear command 52
clear text 134
clicking in spectrograms 231
clicks in mini-display 111
clicks in signal display 111
clicks in spectrogram/cepstrogram 111
clicks in spectrum/cepstrum 111
clicks on information bar 112
clicks on signal control panel 113
co-processor use 145
Coller 133
color 8
color and grayscale spectrograms 14
color indicator 112
combination buttons 154
combined signal and calculation memory 214
commas 128
compare two different zooms 41
Complex transformation 191, 193
Complex transformation setup 191
complex transformations 77
complexity 211
Concernant Signalyze 120
cone kernel 57
connected line plot 154
constant 187
contrast setting 149
control panels 24
Copier 132
Copy 132
copy restrictions. 233
copy signal 132
copy text 132
copyright notice 120
cos 188, 190
Couper 132
courbe spline 197
creator signature 52, 124, 144
curseurs 163
cursor manipulation 212
curve fitting 81, 196
Cut 132
cut, copy, paste, clear 74

Darkness 67, 175
darkness adjustment 176
dB 49
dB difference 50, 142, 188, 190, 142
dB differences in an RMS envelope 79
dB differences to the value 1 79
decibel differences 15
decibel values from a spectrum 62
decibels 177
decimal point/comma 49, 157
default label options 47
Delete all but first 161
Delete all signals 161
Delete signal 161
Delete signal display 160

Delete signals on card 161
delete signal/display button 118
delete the display as well as the signal 160
deletes the display 160
deleting a signal 41, 228
delta 49
delta 141, 142
department 233
departmental and organizational licenses 233
derivative differences 192
derived signal 152, 155
design considerations 210
destination track 117
destination track button 69, 70, 117
dialogues de précaution 162
different languages 103
Digidesign boards 159
Digidesign devices 208
digitization 206
Direkteingabe 209
disk 105
display setup 153
display size button 117
displays 31, 41, 212
dithering a spectrogram with a graphics program 97
divide 189
don't read manuals 228
dot matrix 99
dot plot 153
downsampling 229
Drucken 131
Druckereinstellung 131
duration 30, 50, 143
duration indication 45, 140
duration indicator 30, 112
duration measures 42

Edit 73
edit buttons 115
edit window 23
edit window 44
Edition automatique 203
Edition du signal 134
EEG 18
Effacer 133
Einfache Umwandlung 191
Einsetzen 133
EMG 18
enable button 8, 117
enable delete-save dialogs 162
end of file 224
end point values 142
English 103
Enlever affichage du signal 160
Enlever signal 161
Enlever signaux de la carte 161
Enlever tous les signaux 161
Enlever tous sauf premier 161
Envelope 195
Envelope setup 193
Enveloppe d'amplitude 195
Envoyer à la fenêtre de texte 143
ergonomic considerations 210
Etablir nouveau début 162
Etablir nouveau MinMax 162
Etiketten zeigen 151
Etiketteneinstellung 150
every pixel 154
extension codes for new signals 236
external synthesizers 158
extra wide band 171, 55
extract pitch 169
Extraction de fréquence fondamentale 169

Fakultative Menüs 185
Fast Fourier Transform 54, 176, 166, 229
fast envelope 193
Fenêtrage droit 135
Fenêtrage gauche 135
Fenêtre de signal 186
Fenêtre de texte 186
Fenster verbergen 185
FFT 54, 176, 166, 229
FFT-comb 64, 167
FFT-comb analysis 167
FFT-comb routine 165
File menus 122
File setup 127
file formats 89
file length 152
FileConverter 89, 224
Filter 201
Filter setup 199
Filtering 76, 83
Filtrage 201
Find in signal 136
Find in signal again 138
Find label again 139
find 137
finite vs. infinite impulse response filters 84

FIR (finite impulse response) filters 200
FIR vs. IIR filters 84
first and second derivatives 81
first and second signal buttons 69
floating point operations 226
floating-point 174
font 138
fonts and font sizes 52
force 16 bits 230
formant spectrogram with LPC 64
formant spectrum via LPC 63
French 103
frequency equivalent 31
frequency indicator 112
frequency precision—frequency range tradeoff 176
frequency range 165
frequency/bit width indicator 112
frication 82, 198
frication index with zero crossings 82
FTP 108
full range 172
fundamental and its harmonic frequencies 56
fundamental frequency 166

General control panel 23
general control panel 113
general functions 211
generic format 130
generic sound resource format 222
German 103
Gesamtes Signal zeigen 163
Geschwindigkeitspalette 186
get a specific numeric value 42
Gewählten Signalteil zeigen 163
gopher server 108
graphics files 96
Graphik 184
grayscale 174
grid lines 156
Grundfrequenz 169
guided tour 21

Half range 172
Hamming window 179
hard disk 232
harmonic frequency 181
headerless format 223
Help 120, 121
help page 121
Hide all windows 185
Hide on context switch 185
Hide signal window 184
Hide text window 185
high frequency components of a signal 78
high resolution and color 100
high resolution spectrograms 67
Hilfe 121
hints 120
horizontal cursor 156
horizontal line 156
horizontal line button 118
HyperCard stack 125, 130
Hz 49
Hz difference 50, 142

I/O devices 37
icon 21
IIR (infinite impulse response) filters 200
IIR filters 85
Imagewriter 99, 131
import /export 223
import 53
import/export format 129
Imprimer 131
information area 25
information bar 23, 25
information bar 24
Informations sur signaux 152
InfoSignal Inc. 232
ink jet 99
Ins Textfenster senden 143
installation 105
installation suggestions 104
interactive pitch extraction 88
interactivity and speed 213
international characters 53
international keyboards 26
interrupt sound reproduction 160
intonation contour 166
inverse color 67, 149, 175
inverse filter 173

Japan 148
Jouer macro séquentiel 205
Jouer signal 160

Key presses at startup 111
key-click combinations 66
key-click combinations 110, 237
keyboard and clicking actions 109
kinematic (movement) signals 81
knot width 81, 197
Komplexe Umwandlung 193
Kopieren 132
kurtosis 50, 142

Label color 150
label display 151
label file format 225
label files 47
label levels 46, 150
label setup 150
label's name 138
labeling 8, 46, 138
labels 45
Language 148
languages 103
Langue 148
large screens 43
large spectrograms 231
laser printer 99
Laserwriter 99, 131
learning effort 211
left and right end point values 49, 142
left channel 159
levels 8
license agreement 21, 232
limit 192
limited guarantee and warranty 234
linear predictive coding 56
loading a file 28
loc 152
location 155
lock cursors 137
log 67
log scale 174
log10 188, 190
logn 188
long-term averaged spectrum 65
Lösch/Speicher-Dialoge 162
Löschen 133
low pass 174
LPC 54, 56, 171

MacAdios devices 207
MacAdios™ a/d board 35
MacAdios™ board 33, 159
MacAdios™ model 411 33, 35, 159
Macintosh microphone 34
Macintosh Plus/SE 104
Macintosh sound 36
Macintosh sound input 34
MacRecorder® 34
MacRecorder® and similar devices 207
macro menus 204
macro processes 88
MacSpeech Lab format 129, 218
Make plot 184
manual scoring 18, 38, 42
manual window sizing 148
maximum 49, 141, 142
maximum and minimum 183
maximum number of signals 148
maximum value 42
mean 50, 142
measure amplitude with an RMS envelope 78
memory 102, 107, 120, 128
memory allocations 147, 214
memory management 214
memory request 106
memory under System 7 147
memory-greedy 228
menu bar 148
menu color coding 122
menu commands 110
Menus optionnels 185
menus 120
microsecond standard 226
mike technique 209
millisecond 8
mini display 23, 29
minimum 49, 141, 142
minimum and maximum values 72, 152
MinMax 31, 163
MinMax by clicking 71
MinMax display 153
MinMax via the information dialog 71
missing values 41, 141

modem/printer port 207
modifier keys 26
money-back guarantee 107
mono 159
Montrer segment sélectionné 163
Montrer signal entier 163
more on memory 147
more signals 152
move signal 119
move with key-clicks 31
move with scrollbar 31
moving a signal to another track 40
moving average 86, 200
moving left and right 31
MultiFinder 106
multiple signals 69
multiplication 189
multiply 189
music 19, 171

N valid observations 50, 143
narrow and wide signal displays 39
narrow band 54, 171
narrow-band spectrum /cepstrum 60, 61
Neuen MinMax 162
Neuen Signalbeginn 162
Neuer Text 122
Neues Signal 122
New 52
New MinMax 162
New Signal 122
New Start 162
New Text 122
NIH image 96
Nochmals Etikette suchen 139
Nochmals im Signal suchen 138
noise or echo component of a signal 76
Nouveau signal 122
Nouveau texte 122
novelty 211
Nulldurchgänge 199
number of decimals 49, 157
number of observations 50, 143
number of repetitions 159
numerals 45
Numeric setup 157
numeric formats 48, 226
numeric scales 156
numeric type 77, 128, 192
numeric type in ASCII format 95
numerics 224
Numériser 209
Nyquist frequency 54, 84

Observe 197
offset bias 77
offset bias 192
one-shot transfer mode 73, 115, 133
Open image 126
Open signal 122
Open sound resource 130, 125
Open text 123
Optional menus 185
Optionen Amplitudenhülle 193
Optionen Dateitypen 127
Optionen Direkteingabe 206
Optionen einfache Umwandlungen 187
Optionen Etikettensuche 138
Optionen Filterung 199
Optionen Graphik 182
Optionen Grundfrequenzanalyse 164
Optionen komplexe Umwandlungen 191
Optionen Maßstab 156
Optionen Nulldurchgänge 198
Optionen numerische Formate 157
Optionen Serienmakro 204
Optionen Signalbearbeitung 134
Optionen Signaldarstellung 153
Optionen Spektralanalyse 169
Optionen Splineglättung 196
Optionen Wertübertragung 139
Options d'affichage 153
Options d'analyse spectrale 169
Options d'échelle 156
Options d'édition automatique 202
Options d'étiquetage 150
Options d'impression 131
Options de démarrage 146

Options de fichier 127
Options de filtrage 199
Options de fréquence fondamentale 164
Options de graphique 182
Options de macro 204
Options de numérisation 206
Options de recherche par étiquette 138
Options numériques 157
Options pour affichage de statistiques 141
Options pour affichage de valeurs 139
Options pour audio 158
Options pour courbes spline 196
Options pour enveloppes d'amplitude 193
Options pour passages par zéro 198
Options pour transformations complexes 191
Options pour transformations simples 187
order (coefficient) number 57, 171
order 171
order number 56
organization 233
Original MinMax 163
Original start 162
output filter 165
output tolerance 93, 127
Ouvrir image 126
Ouvrir ressource sonore 125
Ouvrir signal 122
Ouvrir texte 123
overflow 188
ownership of software 233

Page setup 99
Page setup 131
palette de vitesse 186
palette spectrographique 149
palettes 149
Passages par zéro 199
Paste 133
paste text 133
pattern 183
pause length 144
peak-to-peak envelope 193
peak-to-peak period detectors 167
phase plot 184
phonetic font 9, 138, 150
PICT image 126
Pitch extraction 166
pitch analysis 55
pitch detectors 166
pitch extraction button 114
pitch extraction setup 164
pixels drawn on x-axis 154
pixels drawn on y-axis 153
Play sequential macro 205
Play signal 160
play 8
play all but the selected segment 32
play back menu command 87, 113, 114
play the selected segment 32
play the whole signal 32
Plot setup 182
plot every line 175
plot every other line 175
plots 182
plotting 153
points in the signal 8, 47
portable and PowerBook adaptations 213
portions of the signal 8
Power Macintosh 10
Power Macintosh 104
power 189
power and RMS envelopes 194
power envelope 78, 194
pre-emphasis 173
Preferences 144
Préférences 144
principles of key-click assignment 110
print 131
print title 175, 183
printer settings 99
printing 99, 131
printing a high-resolution spectrogram on a separate window 100
printing a plot window 101
printing a signal display with a low-resolution spectrogram 99
printing a text window 101
programmability 214
programmable buttons 204
protective dialogs 161

Quarter range 172
quefrency 56

questions 228
Quit 132
Quitter 132

RAM Doubler 16, 107, 229
range 172
ratio 49, 141, 142
read-only operation 145
Recherche de signal 136
Recherche de texte 137
reciprocal 188, 190
recommended hardware 104
Record 11
Record setup 206, 209
record menu command 87, 113, 87
Reestablish the display 160
reference minimum 80
reference section 109
reg 71, 152
regular signal 152
repeat every other line 175
repeat function 109
replace with 137
resize button 183
resize the signal 162
resolution 65, 174
Rétablir début original 162
rétablir MinMax original 163
return key 110
right channel 159
RISC processor adaptations 214
RISC-based machines 104
RMS 79, 194
root mean square 194

S.d. (standard deviation) 50, 142
sample at cursor 25
sample number 45, 140
sample value 45
sampled display 153
samples examined 153
sampling frequency 72, 77, 191
sampling rate 57, 95, 152, 171, 207, 224, 123
Sauver comme ressource sonore 125
Sauver image 126
Sauver signal 123
Sauver signal sous 124
Sauver texte 123
Sauver texte sous 124
Save as sound resource 125
Save dB differences 81
Save dialogs 161
Save image 126
Save signal 123
Save signal as 124
Save sound resource 130
Save text 123
Save text as 124
saving the text file 52
scale limits 14
scale setup 156
scales 156
scales and/or gridlines 175
scales button 115
scrap 227
scroll bars 119
Search by label 138
search for labels 9
searching for labels 47
second signal 187, 228
second signal button 116
Select all 134
Select all text 52
selected portions of the signal 47
selected segment 159
selected signal button 116
selecting a signal track 28
selecting portions of the signal 29
Sélectionner tout 134
send to text window 143
separator 95, 128, 224
Sequential macro setup 204
sequential macros 88
sequential processes 205
Serienmakro 205
shift left 189
shift right 190
shift-clicking 30
Show statistics 143
Show statistics options 141
Show values 45, 140
Show values options 44, 139
Show whole signal 163
show labels 151
show selected segment 163
sideways display 99, 101
Signal löschen 161
Signal öffnen 122
Signal speichern 123
Signal speichern als 124
signal 17
signal acquisition 105
signal analysis 17, 38
signal and calculation memory 102

signal control panel 23, 25, 116
signal displays 211
signal edit 133, 134
signal editing 69, 73
signal editing mode 73
signal except for selection 159
signal file formats 89, 127, 216
signal file input and output 93
signal information 71, 152
signal length 71
signal memory 102, 148, 215
signal menus 152
signal name 25, 72, 224
signal placement 70
signal replot button 118
signal selection 39
signal selection button 117
signal splice 135
signal statistics 49
signal window 23, 186
signal window layout 211
signal-specific functions 211
Signaldarstellung löschen 160
Signale dieser karte löschen 161
Signalfenster 186
Signalfenster verbergen 184
Signalinformation 152
signals 41, 212
signals stored as files or as resources 92
Signalsuche 136
Signalwiedergabe 160
Signalyze format 129, 217
SigUpdater 108
Simple transformation 191
Simple transformation setup 187
simple transformations 76
sin 188, 190
single-button mouse 212
single-user (individual) license 232
skewness 50, 142
slice plots 181
slim band 171
slope 49, 141, 142
slow-down 10
smooth a signal splice 74
smoothing 174, 177
smoothing at the transition 135
software license 232
Sound Designer II format 129, 219
Sound Manager 3.0. 36
Sound Manager 36
sound accelerator 35
sound driver 36
sound input 32, 34
sound level 159
sound output devices 33
sound playback button 118
sound reproduction 32, 160
sound resource 93, 125
sound resource format 127, 221
sound resources 130
SoundEdit / MacRecorder format 129, 219
SoundEdit 232
SoundWave / Impulse format 129, 220
source signal/new signal after transformation 146
space as a separator 95
space character 224
spacing 175
spectra 181
Spectral analysis setup 169
spectral analysis 18, 54, 176, 178
spectral frequency and amplitude (intensity) 177
spectral sections 181
spectre/cepstre 182
spectrogram colors 66
spectrogram darkness 66, 112
spectrogram generation 181
spectrogram on separate window 63
spectrogram on signal window 62
spectrogram palette 66, 149
spectrogram settings 68, 174
spectrogram-cepstrogram 182
spectrogram/cepstrogram button 114
spectrogramme-cepstrogramme 182
spectrum/cepstrum 182
spectrum/cepstrum button 115

speed modification 202
speed palette 10, 23, 186
speed-up 10
Spektrogramm-Cepstrogramm 182
spektrographische Palette 149
spektrum/cepstrum 182
Spline setup 196
spline 197
Splineglättung 197
splines 18, 81, 196
Sprache 148
spreadsheet programs 53
sqrt 188, 190
standard formats 130, 221
Starteinstellung 146
Startup options 102, 146
statistics on a pitch extraction 50
statistics programs 53
Statistiken übertragen 143
stereo 127
stop 33
stop execution 114
stop playback 33
stop with mouse 207
stopping spectrogram 176
store key for menu playback button 146
subtract 189
Synchrone Cursor 163
synchronous cursors 69, 163
System 7 147
System 7-compatible device 34

Tab character 45
tan 188, 190
Taper left 135
Taper right 135
tapering 74, 135
technical information 210
temporal structure analysis (TSA) 54, 60, 64, 164, 167
tension spline 196
terms 22
Text öffnen 123
Text speichern 123
text 223
text edit 132, 134
text editing mode 73
text editors 227
text files 96, 123
text scrolling speed 146
text search 137
text search and replace 51
text speichern als 124
text window 23, 48, 186
text window width 52, 146
TEXT/ASCII format 128, 129
Textfenster 186
Textfenster verbergen 185
Textsuche 137
threshold 165
time calculations 226
time indication 45, 140, 157
time indicator 112
time locking 69
time plot 183
time-stamp 8, 47
title 45
title and sampling rate 128
titles 139
titles used in transfer operations 139
Toneinstellung 158
Tonressource öffnen 125
total n observations 50, 143
total number of numerals shown 157
transcendental transformations 188
Transform menus 187
Transformation complexe 193
Transformation simple 191
transformations 76
transition noise 203
transition smoothing 74
trimming 202
Trouver à nouveau 138
Trouver étiquette à nouveau 139

Über Signalyze 120
Undo 231
unrelocatable memory block 216
up and down arrow keys 109
update policy 234
updates 108
Ursprünglichen MinMax 163
Ursprünglichen signalbeginn 162
user support 107

Value greater than a given value 43
value of sample at cursor 25
velocity 196

verification of pitch extraction results 168
vertical line cursor 212
vertical line plot 154
vertical zoom 10, 40
vertical zoom and MinMax setting 70
very narrow band 171
very slim band 171
very wide band 171
video card 229
video processing 229
virtual memory 107, 229
virus 230
visible tabs 45
visible tabs 139
voice 166

Wahleinstellungen 144
warranty 234
Werte übertragen 140
whole signal 159
whole/decimal 224
wide band 54, 171
wide band spectrograms 67
wide-band spectrum of a vowel 61
window and palette layout 211
window overlap 181
window sizing, cards and placement 148
windows 22
word processing programs 53
word-wrapped 227

X-Y plot 183

Zero crossing setup 198
Zero crossings 82, 199
zero baseline 72, 128,190
zoom scroll bar 119
zoom signal 119
zooming 28
zooming with key clicks 29
zooming with menu commands 28
zooming with scroll bar 29
Zuspitzen links 135
Zuspitzen rechts 135