# WaveTrak Cards

A blank WaveTrak stack, that is, one without any user data stored in it, consists of a number of cards that perform a variety of important functions. This chapter presents a detailed description of each type of card in a WaveTrak stack. The card names as far as HyperCard is concerned (i.e. the names entered in the 'Card Info...' box for each card) are usually abbreviated from those used in the manual. The table at the end of this chapter gives you the HyperCard names for each card; you will need these names when referring to cards from within your scripts.

*Home Card*

When you first launch WaveTrak and dismiss the opening screen by clicking the mouse, you are taken to the first card of the stack, called the 'Home Card' (not to be confused with HyperCard's Home *stack*). A screen dump of a typical Home Card is shown in Fig. 6-1.

The Home Card gives you general information about the stack, such as the number of roots, traces and cards, as well as the size of the stack in kilobytes.

Tip:

HyperCard stacks tend to accumulate excess bytes as they grow. Compact your WaveTrak stacks periodically by selecting the 'Compact Stack' menu item under the 'File' menu. Compacting may take several minutes for a large stack. You should always compact a stack once you're finished adding traces to it and plan to archive it.

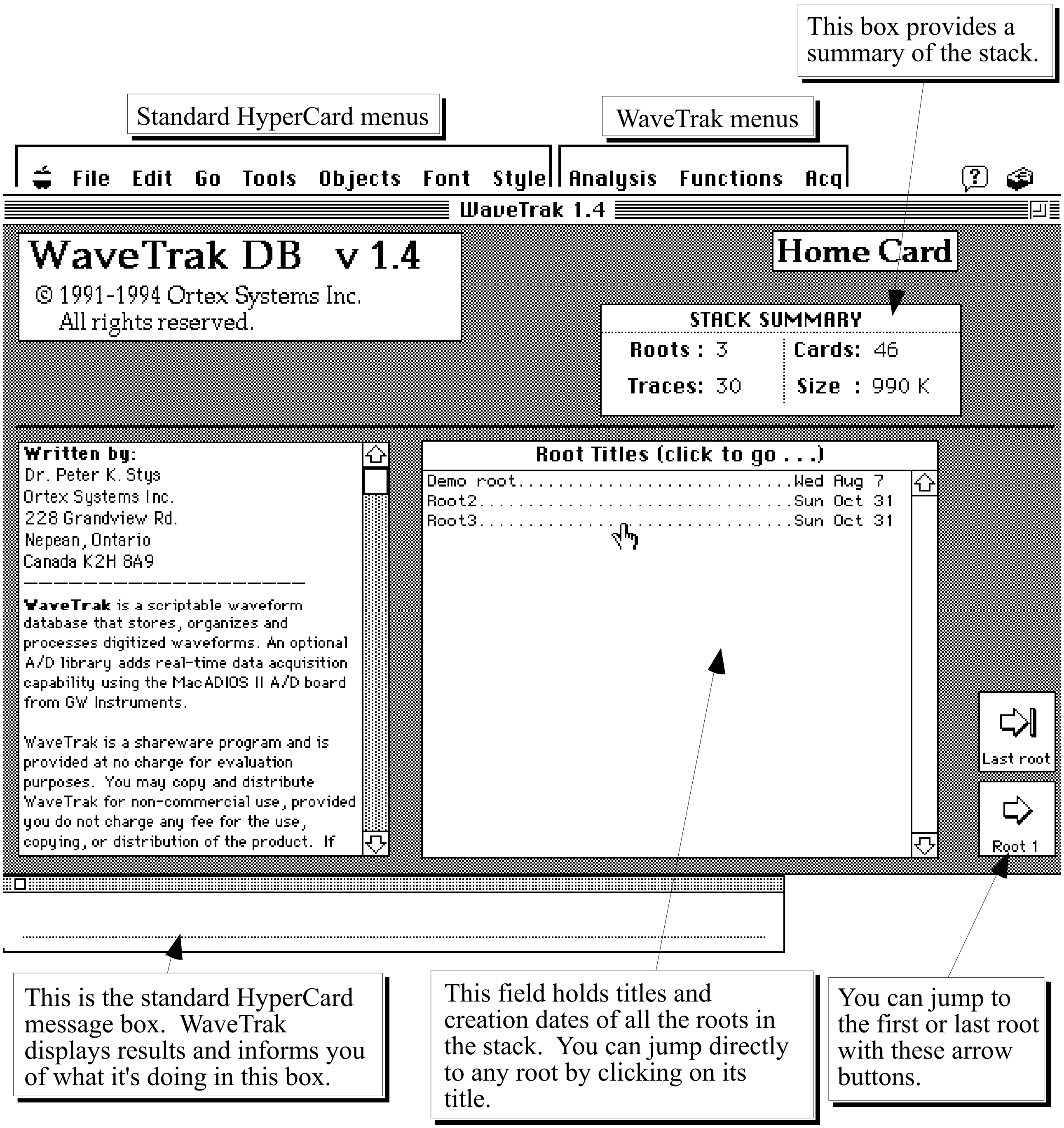



Fig. 6-1: Screen dump of the Home Card, the first card in the WaveTrak stack. This card contains fields holding general information about the stack, such as the number of roots, traces and cards in the stack, as well as a listing of all the root titles. The top left will read 'WaveTrak DB' or 'WaveTrak AD' depending on whether you have the optional A/D library installed.

If you have one or more roots in the stack, two arrow buttons at the bottom right corner of the Home Card are used to jump to either the first or last root in the stack. These buttons will not be visible if there are no roots in the stack, such as when you run WaveTrak for the very first time. If you accumulate many roots in a stack, it becomes awkward to sequentially move from root to root using the arrow buttons. The 'Root Titles' field in the Home Card lists all the titles from each root in the stack, similar to a table of contents in a book. Clicking on a title will send you directly to that root.

You can always get back to the Home Card from anywhere in the stack by selecting the 'Home Card' menu item under the 'Go' menu, or by typing Command-1 (⌘-1). The latter is the standard HyperCard keyboard shortcut for jumping to the first card in a stack. This is the only keyboard shortcut we recommend. *Do not use the 'Go Prev' or 'Go Next' commands* since the sequence of cards as far as HyperCard is concerned is different from the logical organization of roots and traces in WaveTrak. Use only the commands provided under the Go menu or the navigation arrows.

Think of the Home Card as a "home base", to which you can jump any time to get your bearings should you get lost. You will find, however, that the stack structure is simple and intuitive enough that it won't take much time to learn how to get around without difficulty.

## *Root Card*

The first card of every experiment is the root card. It serves as an introduction to the experiment and contains a synopsis of results, much as the Home Card served as the introduction to the stack. A sample root is shown in Fig. 6-2.

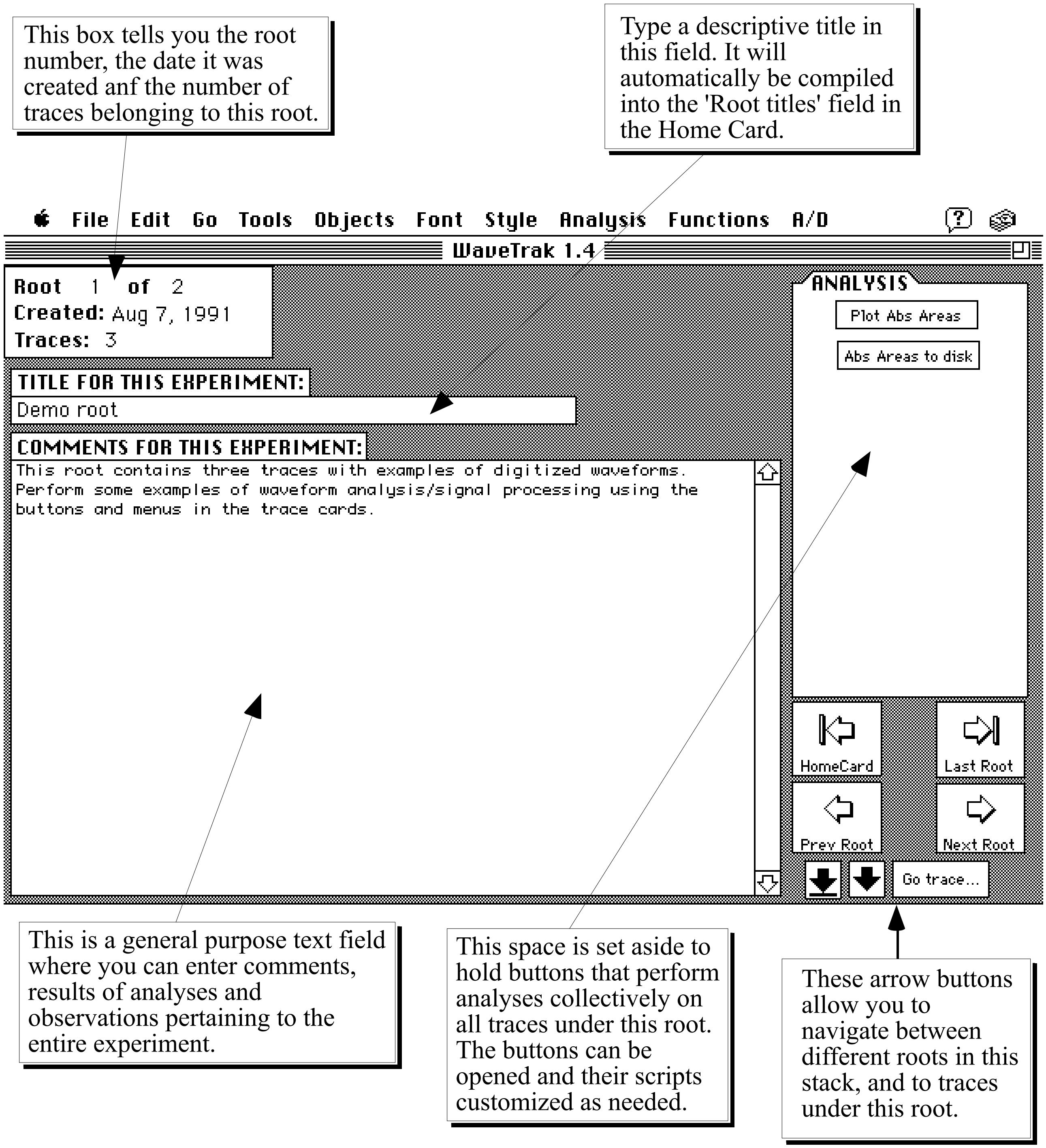



Fig. 6-2: Screen dump of a root card, which is always the first card of each experiment. The root card holds general information about an experiment such as the creation date, how many traces are contained under it, a field for comments, and buttons to navigate from root to root. Arrow buttons also allow you to go to the first or last trace under this root.

Each root is date stamped when it is created. Roots are numbered sequentially and new roots are always appended to the last existing root. This way, roots are always arranged in chronological order. The number of traces belonging to this root is shown in the upper left. You can enter a descriptive title for each root in the title field.; this title, as well as the creation date, will be automatically compiled to the 'Root Titles' field in the Home Card.

Below the title field is a general purpose comment field where you can type notes, results of analyses or observations pertaining to the experiment; this field functions as the first page of an experiment in a traditional laboratory notebook. As you get more advanced, you can write scripts that perform automatic analyses on your data under a root, and append the results to any notes you typed in this field. The Comments field thus functions as a summary of an experiment, for quick review. The Title and Comment fields are compiled into the WaveTrak Summary stack, which can hold a synopsis of hundreds of experiments in relatively little memory by excluding the raw data.

Technical note:

Even a blank stack contains a single root card, named 'StdRootCard', from which all new roots are cloned. It serves as a model from which to create new roots. This template is not linked into the normal sequence of roots so you normally never see it. You can change the template by changing any normal root *in the background*. Subsequent roots will reflect your new changes.

## *Standard Root Comment Card*

A default root comment is copied from a field in the 'Standard Comments Card'. You can customize this text by changing the contents of this field in the Standard Comments Card. The section describing the 'System Parameters' card explains how to jump to cards that are not listed under the 'Go' menu.

## *Trace Card*

In WaveTrak, a series of values (either imported, digitized or computed) is called a "wave". However, a series of numbers is meaningless without additional information such as the time between successive elements (i.e. the sampling rate), the gain of any amplifiers before

the A/D converter, whether any stimulating signal was applied to evoke the signal, temperature, and so on. A wave with this additional information is called a "trace", and a single trace is stored on a single 'Trace Card'. Thus, a trace is not just a digitized wave, but a complete record of an acquisition, including hardware parameters, user comments and time stamp. Traces acquired over the course of an experiment are stored "under" their root, and, like root cards, are numbered sequentially and arranged chronologically. See the WaveTrak Organization chapter for a detailed description of the relationship between the Home, root and trace cards.

As each trace is imported or acquired, it is numbered sequentially and time stamped with the current time-of-day (upper left corner, see Fig. 6-3). To the right of this box is a scrolling comment field where you can enter notes about each trace. The up and down arrows immediately to the right of this field will skip over empty comment fields and jump to the previous and following annotated trace cards, respectively. The button with the question mark will prompt you for a text string and search *down* from the current trace card. Once a search reaches the last card, you are given the option of searching again from trace 1.

The 'Show Wave Info' button opens up a pop-up field with four sections displaying information about the acquisition. This is discussed in more detail below.

At the top right corner is another pop-up field which you can use to mark your trace. For example, let's say you are studying four specimens during the same experiment, and you take readings on all four specimens periodically over several hours. Rather than creating separate roots for each specimen, you could mark all traces acquired from the first specimen with 'A', from the second with 'B' and so on. Even if traces from the same specimen may not be contiguous, the marks can be used to indicate which traces belong to the same subject and should be analyzed together. The first trace defaults to 'A'. If you change a mark, all subsequent traces will inherit the new mark. An 'X' is provided for flagging traces that should be ignored. If you change the mark of the last card to 'X' to exclude it, remember that the following and all subsequent acquisitions will also be marked 'X'. Therefore, make sure you manually reset the mark of the acquisition following the 'X'.

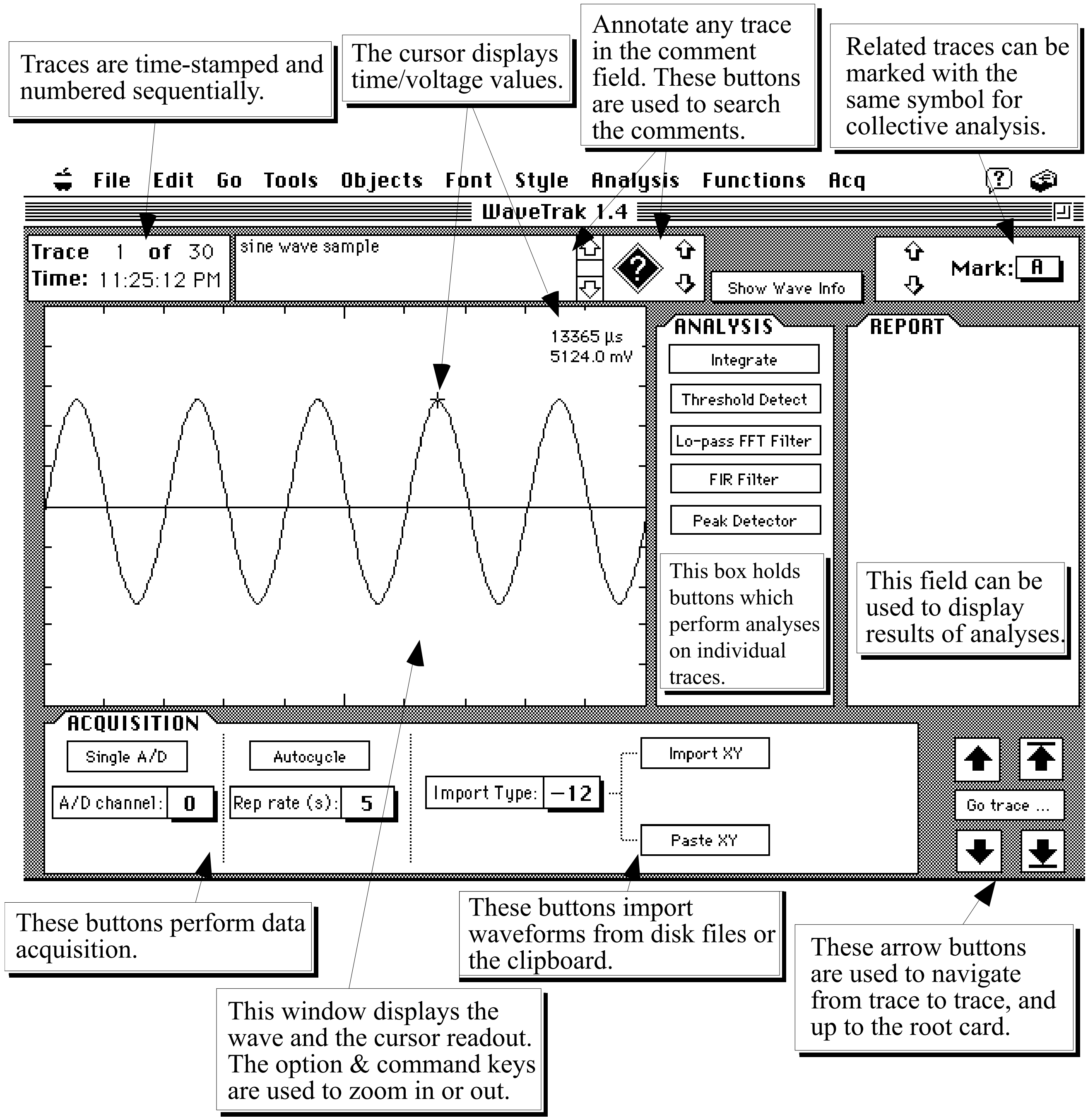



Fig. 6-3: Example of a typical trace card. Digitized waves, along with other information pertaining to the acquisition are stored on trace cards. In addition, trace cards hold buttons for acquiring new signals, performing analyses and displaying waves in the display window.

Technical note:

The displayed time stamp is actually computed from a complete time and date value stored in a hidden background field named **capturedSecs**. This value originates from a call to the HyperCard function `the seconds` (e.g. `get the seconds`), which returns the number of seconds elapsed since midnight of Jan. 1, 1904 and the current time set on your Macintosh. The value you see in the top left corner is computed from the **capturedSecs** field and displayed as a 12 hour time. By storing the complete time, you can determine not only the time-of-day but the date of capture of any trace if your experiment should span more than one day. Examine the `on newTrace` handler in the trace background for details on how the two times are stored and converted. Your HyperCard reference will explain the HyperTalk functions used to convert between various time formats.

Below, the 'Analysis' box is an area where buttons are kept which perform various analyses and manipulations on traces. When you add your own buttons, *make sure to place them into the background*, or else they will only appear on a single trace card. See the HyperCard manual for an explanation of background/foreground layers. Similarly, the 'Report' field is a general purpose field where analysis buttons can display their results, since the message box allows only one line, and data must frequently be shown in tabular format.

Navigation buttons near the bottom right corner allow you to move from trace to trace, directly to the root card, or to the last trace card, as shown in Fig. 4-7. Holding the mouse down on an arrow button will continuously move you from trace to trace. The up and down arrow *keys* will go to the previous and next trace cards, respectively. Holding down the shift key will jump ten traces at a time. The 'Go trace...' button allows you to jump directly to any trace card by entering its number in the dialog box. You cannot move from a trace card under one root directly to another trace card under a different root.

The 'Acquisition' box near the bottom of the card holds buttons for performing signal acquisitions, either by performing analog-to-digital conversion with the required hardware, or by importing pre-acquired data from other files or the clipboard. The default trace card comes with several buttons:
'Import XY' and 'Paste XY' will import data from a disk file or the clipboard, respectively, create a new trace card to store the data, and link the new trace card into the database for easy navigation. These two buttons are virtually identical, except for the source of the data (disk file or clipboard), so they will be discussed together. Their main function is to read in an ASCII table of numerical values, convert these into WaveTrak's compressed format

and store this *wave* in a new trace card. In addition, various fields are provided to more explicitly describe the acquisition (made visible by the 'Show Wave Info' button). The ASCII table can be either a series of Y-values only, separated by carriage returns, or a series of X,Y values, delimited by a comma, tab or any number of spaces, with each XY pair terminated with a carriage return. Fig. 6-4 shows some sample data:

```
-363.9          0,-363.9
-290.6          10,-290.6
-207.6          20,-207.6
-134.3          30,-134.3
   .               .
   .               .
   .               .
5277.2          50920,5277.2
5296.7          50930,5296.7
5306.5          50940,5306.5
5326.0          50950,5326.0
5335.8          50960,5335.8
```

Fig.6-4: sample data illustrating format of data containing ASCII Y-values only (left column) and XY comma delimited pairs. Note that every line in each of the two sets ends in a carriage return. Note also that X values must be evenly spaced (either assumed to be in left data set, or explicitly shown in right set).

When you import the set of Y-values, WaveTrak cannot know what the intended X interval is between successive samples. In this case, X values are determined from the current sample rate setting in the Scope card. Before importing your data, therefore, you must manually preset the sampling rate in the Scope card. All subsequent acquisitions will use this setting until you change it. WaveTrak always assumed however, that *samples are evenly spaced in time* (or any other dimension).

---

Tip:

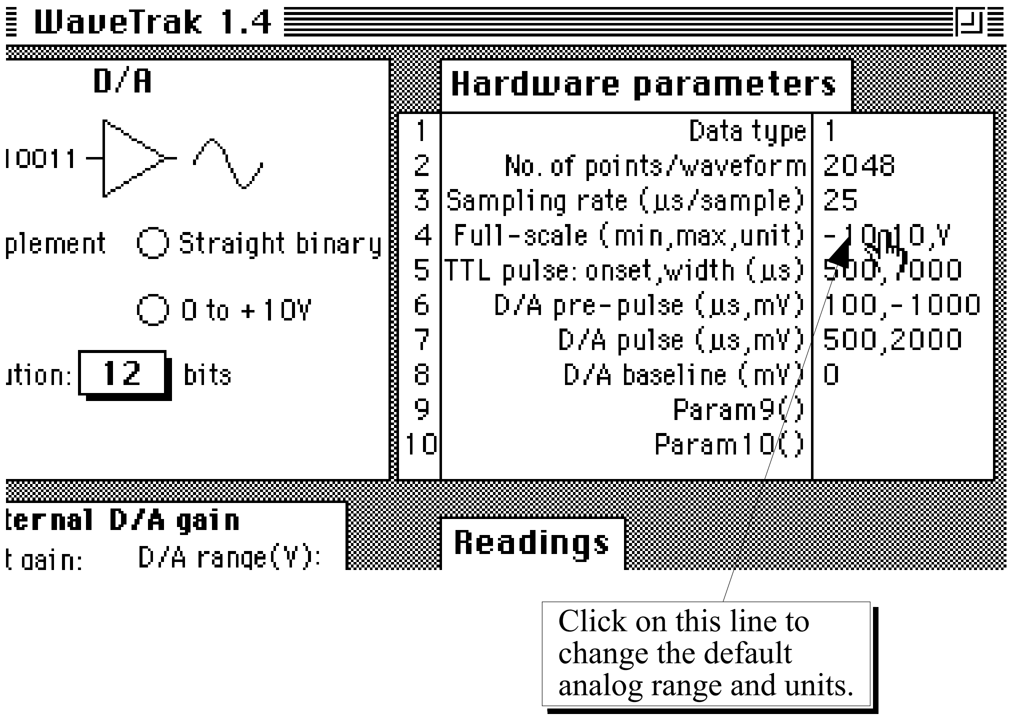
Although 'sampling rate' settings for X-values imply time as the X dimension, this need not be the case. You can enter any unit you wish to be used as the default in line 3 of the Hardware Parameters field in the System Parameters card.

---

If instead you have a table of XY pairs, WaveTrak will ignore settings in the Scope card because it can directly determine the sampling rate from the imported data. This is done simply using the difference between the first two X values. All subsequent x values are ignored but must be supplied so that WaveTrak correctly extracts the Y values from the table. WaveTrak automatically detects whether data contains Y or XY pairs, provided each line ends in a carriage return, and delimiters are restricted to commas, tabs or spaces.

Import Type:
This is an important setting for your data to be interpreted correctly. The *type* of a wave in WaveTrak describes whether it contains binary integer ( 8, 12 or 16 bit) or single-precision floating point values. A detailed discussion of *waves* and *types* appears in the chapter on Scripting. We would encourage you to study this important concept before proceeding. The reason for different types is efficiency and the source where most digitized waveforms come from, that is, A/D converters. A/D converters always return a binary integer with a know resolution (number of bits). For instance, a signed 12-bit converter always returns integers in the range of -2048 to 2047. WaveTrak uses this property, along with its compression routines, to efficiently store binary waves using about 1 byte per 12-bit sample. In contrast, floating point waves require over 4 bytes per sample. Binary waves require a few more parameters to accurately describe what these integer samples really mean. Going back to out 12-bit converter example, let's assume that the device was configured so that an analog input of -10 volts returned a converted integer value of -2048, and +10 volts returned 2047. Knowing that the range of integer values (-2048 to 2047) is linearly mapped to the true analog input range of -10 to +10 volts, we can easily compute the true analog input from any integer sample. This is the purpose of the 'Full scale' entry revealed by the Show Wave Info button in each trace (Fig. 6-6). It tells WaveTrak how to determine the true analog values from their digital integer equivalents.

You should now be in a position to correctly select the Import Type for your data.
If your data consists of raw binary integer samples copied directly from an A/D converter, you must manually set the following parameters for your data to be interpreted correctly:
1) select your converter resolution, e.g. if the samples are signed (i.e. positive or negative) 12-bit integers, you would select '-12' from the 'Import Type' pop-up menu, if unsigned 8-bit samples, select '8'.
2) from the above discussion, you will also have to tell WaveTrak how that converter's integer range maps onto your real analog signal's range: do this in the System Parameters card by clicking on the full scale range and entering your true analog range and units in the dialog box:




This range will be used in all subsequent acquisitions, whether imported or using the A/D converter in the A/D version. WaveTrak now knows that any integer element in your wave with a value of -2048 really means -10 volts, and will make these conversions automatically and transparently, whether displaying the data within WaveTrak with the cursor readout, or exporting to other applications. If some of your values are out of range (e.g. greater than 2047 for a 12-bit signed binary wave), a dialog will inform you, but the wave will still be imported; out-of-range values will be clipped. If this happens however, chances are that your data isn't what you expected it to be, or that you set the resolution wrong (e.g. 8 bits instead of 12).

If your source data consists of floating point numbers, integer bits of resolution is irrelevant, however, you will still have to:
1) set the type to 'F', to tell WaveTrak that you are importing a floating point wave, and
2) set the range of data. Here the range is used not to translate your values, but to select the range for the display window, since with floating point, WaveTrak has no way of knowing beforehand what range you data will span (so the display may be too tiny or clipped). The range is set that same way as for integer waves.

**Example 1, importing a signed 12-bit binary wave**:

You have a disk file containing a wave that was acquired with an A/D converter that returned signed 12-bit values. The gain of the converter and pre-amplifiers was set so that the full range spanned ±10 mV. The converter sampled the analog signal at 50 kHz (once every 20 µs). The system wrote the raw integer values from the converter to an ASCII text file with no translation, and without X (time) values.

Before importing, set the true analog range in the System Parameters card by clicking to the right of the 'Full-scale (min,max,unit)' line in the Hardware Parameters field. Enter **-10,10,mV** in the dialog. Also set the sampling rate by clicking one line above and enter **20** (i.e. 20 µs is equivalent to 50 kHz). These will be the defaults used from now on; if your other files were acquired with the same settings you will only have to enter this once. Return to the trace card and select the correct A/D converter resolution from the 'Import Type' pop-up menu ('-12', meaning signed 12-bit binary). Press the 'Import XY' button and select the file from the file dialog. The wave will be imported and shown in the display window. Note that the number of points in the wave, the correct sampling rate and full-scale range will be copied into the Wave Info field to describe your acquisition. Note also that the cursor will correctly display ±10 mV at the extreme top and bottom of the display window.

**Example 2, importing a floating point wave**:

You have a disk file containing a spectrum that was computed on another system. The X-values are given in MHz, evenly spaced at 2 MHz intervals. The Y-values are in dB and range from -120 to 0. The system wrote the data as a tab delimited file of X and Y values. X values began as 0, 2, 4 etc... reflecting the 2 MHz spacing.
Before importing, you need only to tell WaveTrak what the expected dB range is for a neat display. Click on the line next to 'Full-scale (min,max,unit)' in the System Parameters card and enter **-120,0,dB** in the dialog. The "sampling rate" (or more accurately the X-value spacing in this case) will be automatically determined from the file. Return to the trace card and select 'F' from the 'Import Type' pop-up menu to prepare WaveTrak to read in floating point data. Press the 'Import XY' button and select the file from the file dialog. The wave will be imported and shown in the display window. Note that the Y unit and range are correctly displayed. The X unit is still shown as µs because WaveTrak was written for time-series data; you can easily change this by editing the line `put "µs" into Xunit` in the `openCard` handler of the background script (see the Scripting chapter for details). You can customize WaveTrak as much as you want by editing the button and background scripts, all of which are accessible to the user. For instance, you will no doubt want to add additional instructions in the 'Import XY' button to enter other data in the Wave Info fields automatically.

*WaveTrak AD version only:* 'Single A/D', which, as the name implies, acquires a single wave from the A/D channel selected in the pop-up menu underneath. Sampling rate and number of points are set in the Scope card in the SAMPLING RATE window (see the description of the Scope card below). Autocycle simply calls the 'Single A/D' button at predefined intervals, which are shown (in seconds) in its pop-up menu. With the Autocycle feature you can let WaveTrak run unattended overnight or for days in a remote location, then analyze the data collected later.

The best way to learn how these buttons work, is to open their scripts. Comments have been included which should make the scripts very easy to understand. The Button Bank card contains numerous pre-programmed buttons that perform a variety of A/D functions; you can copy and paste them into this box (remember to paste them into the background layer), and you will no doubt want to customize them to suit your application. These and other WaveTrak buttons are described in detail in the WaveTrak Buttons chapter.

Finally, the display window (with tick marks along its borders) is used to display the digitized wave. This window has several important features which allow you to examine your data in detail. When you position the cursor inside this window, the shape changes from the standard HyperCard 'hand' to a crosshair (see Fig. 6-3). The center of the crosshair cursor is translated into time and voltage (or whatever other unit you define during your acquisition), which are shown at top right corner of the display window. Digitized waves usually hold much more information than the screen can display in a single view. You can zoom in to get a closer look at your wave by pressing the option key while the cursor is inside the display window. Its shape will change to an 'expand' cursor. Position the center dot of the expand cursor over the segment of your wave you wish to magnify and click the mouse. The point at the center of the cursor will be translated to the center of the display window, and both X and Y dimensions will be magnified (Fig. 6-5). The cursor readout at the top right will be scaled accordingly. You can continue to zoom in closer and closer onto a segment of interest; magnification is limited to a factor of 256 in either X or Y directions. The magnification factor for each mouse click can be set independently for the X and Y dimensions by choosing 'X mag...' and 'Y mag...' under the 'Analysis' menu. If you want to zoom in only one direction, set the other magnification factor to 1. Always position the cursor over the point of interest; zooming a blank area will work but you will eventually end up with a blank window.

Technical note:

To maximize drawing speed, not all points in your wave are plotted on the screen when your wave is first drawn. One side-effect is that very fast transients (narrow pulses or frequency spectra of very pure sine waves, for instance) may not be drawn at lower magnifications. Zooming in will of course reveal all of the wave's details. The HyperCard global **maxPlotPoints** determines how many points in total will be drawn for any single view in the display window. This global is initialized to 500 at start up. If you expect signals with a lot of sharp transients that you want to see without having to zoom, change the **maxPlotPoints** default in the stack script to a higher number. The 'Display Points...' item under the 'Analysis' menu in the Scope and trace cards lets you conveniently change this global from the menu bar.

To zoom out, press both option and shift keys together. The cursor changes into a 'shrink' cursor. Clicking the mouse while holding these two keys down will zoom out by the same factors, positioning the region under the cursor at the center of the new view. *Double clicking at any magnification will change the view to normal*.

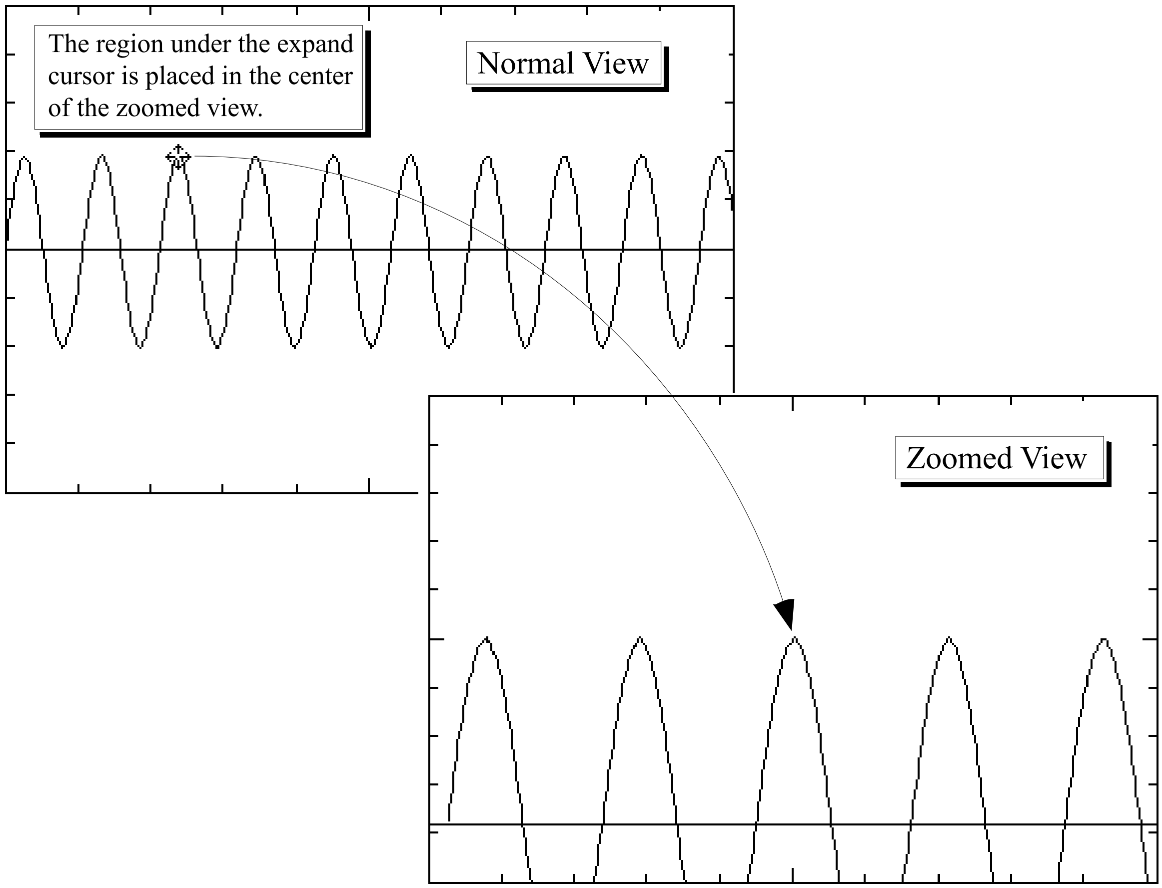



Fig. 6-5: The display window in the trace card allows you to zoom in or out to get a more detailed look at your wave. Pressing the option key changes the crosshair into an 'expand' cursor. Clicking on a segment of your wave will magnify it; the point under the cursor is always placed in the center of the window in the new view. Zooming out is done by pressing both option and shift keys together.

Technical note:

The magnification factor by which the display is zoomed in/out is set independently for the X and Y dimensions. These factors are stored in two HyperCard globals, **xMag** and **yMag**. These globals are both initialized to 2.0 (i.e. with each click, the X and Y dimensions are expanded/shrunken by a factor of 2.0) at startup by the openStack handler in the stack script. You can change the defaults to any value ≥1.0, or you can change the values on the fly from within your own script. For instance, when viewing a frequency spectrum on a log scale, it is usually unnecessary to scale in the Y direction; you can temporarily set **yMag** to 1.0. Note that these globals apply to zooming in any card in WaveTrak, not just trace cards.

The 'Show Wave Info' button opens up a pop-up field displaying information about the acquisition (Fig. 6-6). The upper panel stores *parameters* i.e. information about how the wave was acquired: number of points, sampling rate, A/D gain, whether digital or analog pulses were applied, and so on. Clicking on a line brings up a dialog displaying the full entry for those extending beyond the edge of the field (e.g. the full-scale entry in Fig. 6-6). The bottom panel stores *readings* i.e. measurements taken at the time of the acquisition, such as temperature, baseline DC level, impedance of the recording electrode, and so on. Clicking on a line here allows you to both review and edit the entry. You can hide the 'Wave Info' field by clicking the Show Wave Info button once again.

The contents of this field are copied partly from the default field in the System Parameters card; other entries are written directly by the buttons performing the acquisition. The chapter on scripting with WaveTrak explains where this information comes from. The lines entitled `Param n()` and `Reading n()` are unused and you can store your own parameters or readings here. You can also rename the existing Param and Reading labels to match your data.

Whenever you jump to a trace card, various relevant menu items become active; the 'WaveTrak menus' chapter describes these in detail.

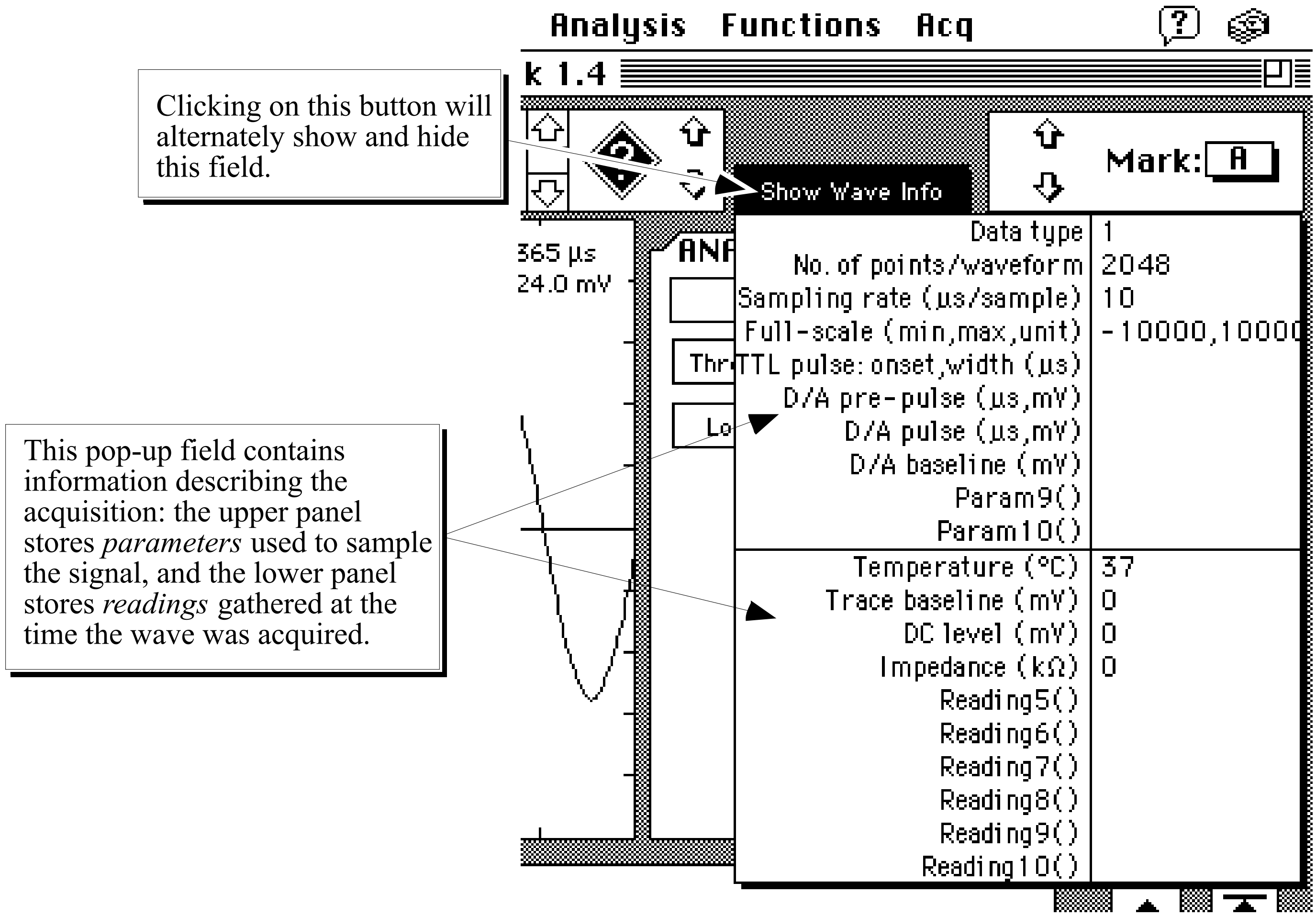



Fig. 6-6: The 'Show Wave Info' button reveals a pop-up field which stores various parameters describing the acquisition. You can view entries in the top panel, and edit the entries in the bottom panel by clicking on each line.

## *Scope Card*

Before actually capturing or importing a trace, it is helpful to preview the incoming signals. The 'Scope Card' functions similarly to the trace cards, except that no new cards are created, and old data are discarded as new signals are acquired. As the name implies, this card functions much like a real oscilloscope. The layout of this card is shown in Fig. 6-7. You can jump to this card from anywhere in the WaveTrak stack by selecting 'Scope' under the 'A/D' menu (or using the Command-

S (⌘-S) keyboard equivalent). This menu item will be dimmed and the scope card will be inaccessible if the hardware check performed at startup failed (you can still get to it from the System Parameters card if you really need to, see below).

The 'SAMPLING RATE' window is used to select the number of points per waveform, the sampling rate and the sampling window time. When you enter two of three parameters, pressing the question mark button beside the third, unknown parameter will calculate it using the other two.

**Example 1**:

You have a disk file containing a wave that was acquired or generated by another system. Press the 'Import XY' button in the Scope card and select the file from the file dialog. The wave will be imported and displayed in the oscilloscope window. If your file consists only of Y-values (ASCII text, one value per line, each ending with a carriage return), you'll notice that the 'No. of points/wave' box under the SAMPLING RATE group will reflect the number of points you read in. The 'Sample Interval/Rate' will remain unchanged because WaveTrak has no way of knowing what the rate was just from a series of Y-values. The 'Sample Window Time' will be calculated according to the number of points imported and the sampling rate you chose. The cursor readout will reflect these settings. If your file contains a series of X,Y values (comma, space or tab delimited), 'Import XY' will automatically determine the sampling rate from the difference between the first two X values. Try importing the file named 'TestWave' provided with WaveTrak; it contains a sine wave of Y-values.

**Example 2** ***(WaveTrak AD version only)***:

Suppose you want to acquire a 50 msec epoch at 100 kHz. Enter 50 ms in the 'Sample Window Time' field, and 100 in the 'Sample Interval/Rate' field (check the 'kHz' radio button). The sample rate can be entered either as a sample interval in microseconds or as a rate in kilohertz. Pressing the question mark button beside the 'No. of points/Wave' field will calculate the number of points needed to satisfy the other two parameters (5000 points @ 100 kHz = 50 ms).

**Example 3** ***(WaveTrak AD version only)***:

You now want to acquire signals as in example 1, but need to acquire a wave with exactly 1024 points so you can perform an FFT (FFT's require that the number of points be an integral power of 2). Again, you want to examine a 50 ms window. Enter 1024 in the 'No. of points/Wave' field, and press the question mark button to compute the required sample rate. The sample rate computed is about 20.48 kHz, or 48.8 μs/sample. Since sample intervals must be in integral microseconds, the real sample interval is rounded to 49 μs, and the 'Approximate' button is checked to indicate that the calculation is not exact. Now click on the Sample Window Time question mark to find out that the true sample window time will be about 50.176 ms.

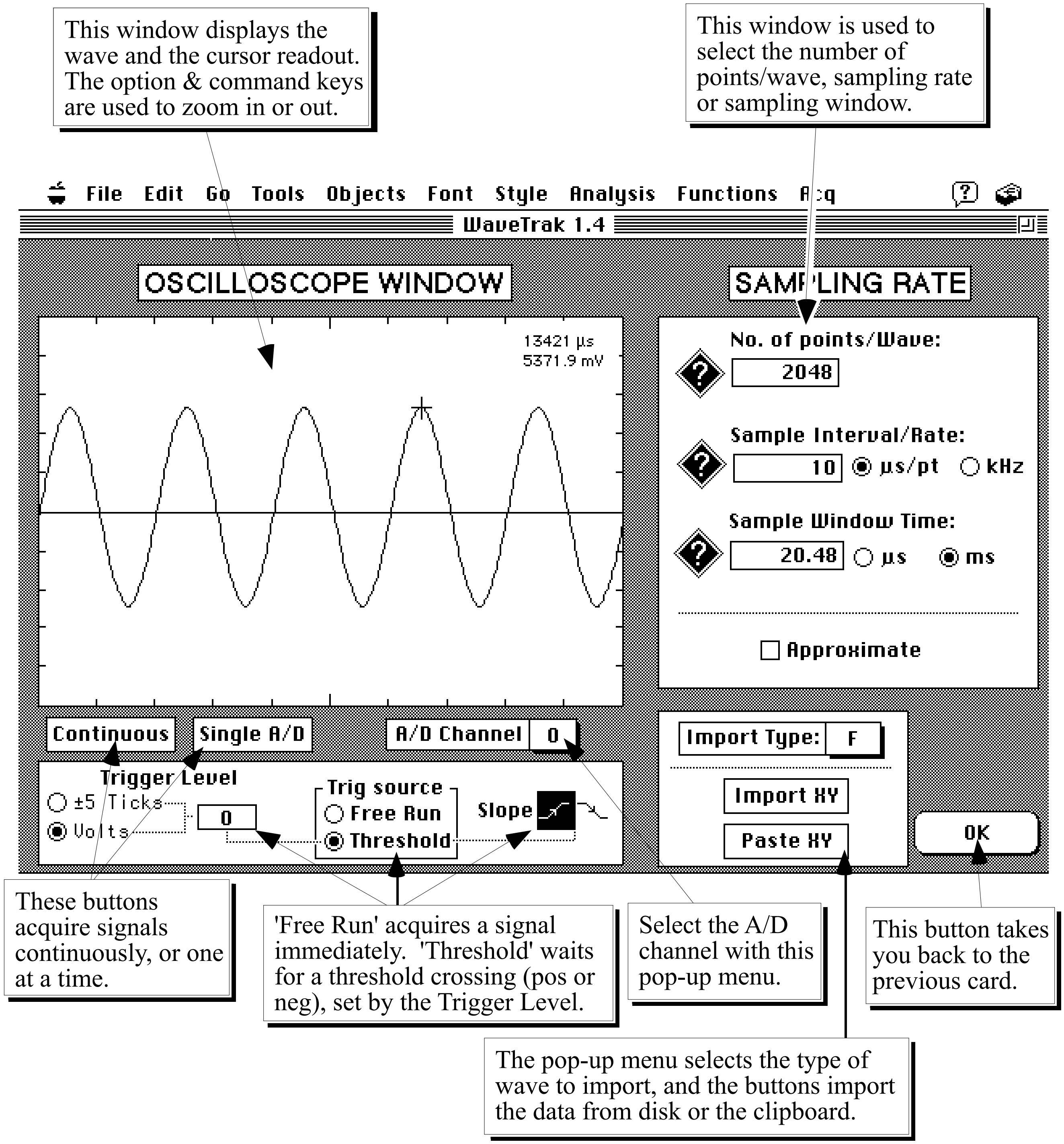



Fig. 6-7: Screen dump of the 'Scope Card'. This card functions like a real oscilloscope, acquiring or importing, then displaying, signals, without permanently storing the data. The settings entered in the 'SAMPLING RATE' will be subsequently used to acquire new traces.

Tip:

Many of WaveTrak's digital signal processing functions rely on the Fast Fourier Transform (FFT). One consequence is that waves manipulated by the FFT must consist of a number of points which is an integral power of 2. We recommend that as a routine, you choose a power of 2 ( e.g., 512, 1024, 2048, etc...) for the number of points in the SAMPLING RATE window. Such waves can be easily processed, without the need for trimming or zero-padding later. If you use WaveTrak's FIR filtering routines, this limitation does not apply.

---

Your new selections will automatically be used to acquire any subsequent traces. The scripting chapter explains how you can access these selections from your own scripts. If you enter new values, but don't press any question marks, the 'Sample Window Time' will be re-calculated automatically from the other two parameters when the cursor leaves the 'SAMPLING RATE' window.

The display window functions exactly as the display window in the trace cards, including cursor readout, and zooming in or out (see above).

*AD version only:*
Below the display window, the 'Continuous' and 'Single' buttons capture signals continuously, or one at a time If you move the cursor into the display window while sampling in the continuous mode, the current wave will be frozen and the cursor readout will be displayed in the top right corner. Sampling will resume automatically when you move the cursor out of the window. Press and hold the mouse button to stop the continuous acquisitions. A signal will be acquired immediately and asynchronously if the 'Free Run' radio button is checked. The 'Threshold' button instructs WaveTrak to track the incoming signal until it crosses threshold, determined by the 'Trigger Level' field. The trigger level is set either as the number of ticks about zero (±5);. this results in a trigger level which is independent of A/D gain, much like the trigger level on a real scope. Alternatively, an absolute level can be entered (in volts) by pressing the 'Volts' radio button. The direction of the threshold crossing can be selected with the 'Slope' button. If no trigger occurs after one second, an error message is displayed in the message box. The baseline is always drawn at zero volts (you can customize this too: see the script for the Single button).
The 'A/D Channel' pop-up menu selects the A/D channel to be displayed.

The 'Import Type' pop-up lets you select the type of wave (floating point, signed or unsigned binary) you want to read in from a disk file (with the 'Import XY' button) or the clipboard (with the 'Paste XY' button). Return to your previous card by pressing 'OK' when you're done. The most recent wave will be saved in the Scope card if it is 30000 bytes in size or less (equivalent to a little less than 30000 points for most 12 bit waves).

## *System Parameters Card*

The 'System Parameters' card contains vital information bridging your external hardware with the WaveTrak software. This card will be most useful for those with the *A/D version* of WaveTrak. You can get to this card by selecting 'System Parameters' under the Go menu. Let's examine each feature by going through the screen dump shown in Fig. 6-8.

The MacADIOS II card is equipped with an on-board 12 bit analog-to-digital (A/D) converter. Analog signals are fed to the converter via a 16 channel multiplexer and a programmable gain amplifier. WaveTrak assumes that your A/D converter is configured for differential input to reduce noise, therefore WaveTrak pop-up menus and tables support only 8 A/D channels (numbered 0 to 7).

> Technical note:
>
> You can access all 16 A/D channels with the WaveTrak toolbox functions.

Both the A/D and D/A converters can be jumpered in a number of ways:

1) coding can be two's complement (i.e. both positive and negative integers) or straight binary, and

2) the full scale can be ±10V or 0-10V (consult the MacADIOS II hardware manual for details).

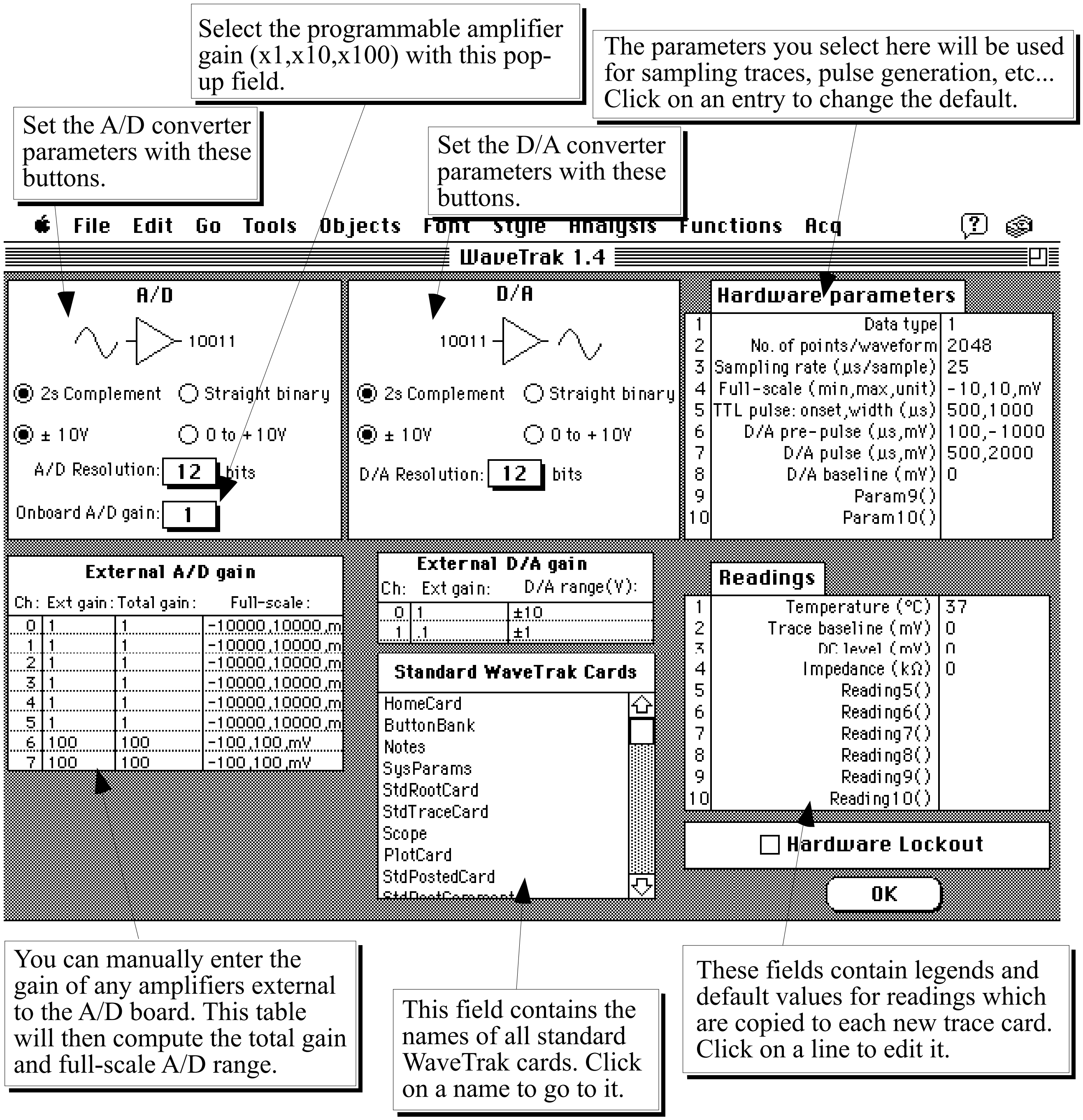



Fig. 6-8: Screen dump of the 'System Parameters' card. This card informs WaveTrak of the current settings and configuration of your hardware, so that digitized signals can be properly interpreted and scaled. See text for details.

The radio buttons under the A/D and D/A icons in the System Parameters card inform WaveTrak of how your board is jumpered. Incorrect data will be generated if these buttons are not set correctly. Only 12 bit A/D and D/A conversion is currently supported on the MacADIOS II card, therefore the A/D and D/A Resolution pop-up menus should be left at 12. The initial radio button settings reflect the factory default jumpering for the MacADIOS II board, so you don't have to change anything unless you re-configure your board.

To successfully translate digital samples into true absolute voltages, in addition to A/D coding and full-scale information, a program must know the total gain of the system from source to A/D input. It is easy for a program to know the gain of the on-board programmable gain amplifier. Frequently though, signals are amplified even before they are fed to the A/D board. WaveTrak has a mechanism by which you can enter the magnitude of any pre-amplification, thus allowing it to compute the true value of a signal: the cursor readout in the trace and scope cards automatically compensates for any gain prior to the A/D converter and always displays true voltage.

**Example**:

You have a signal that is usually confined to a range of ±10 mV. This signal is conditioned and amplified by an external amplifier in your lab which has a gain of x100. Therefore, the range of your amplified signal coming from the amplifier will be ±10 mV x 100 = ±1V. To exploit the full resolution of your A/D converter, set the MacADIOS II programmable amplifier gain to x10 with the pop-up menu, which will give you a final range of ±10V, the full scale input range of the A/D converter (assuming it's jumpered for ±10V operation). You choose to connect your signal to A/D channel 2. In the 'External A/D Gain' table, click on the cell corresponding to channel 2 in the 'Ext. gain' column (third cell from the top), and enter 100, the gain of your external amplifier. You will notice that the corresponding 'Total gain' cell changes to 1000, the product of the on-board gain in the pop-up menu (10), and the external gain you entered in the table (100). The 'Full-scale' column changes to -10,10,mV i.e. the configuration of this channel is such that it can correctly convert a ±10 mV signal without saturating.

Technical note:

Put another way, from a programmer's perspective, the full scale 2's complement 12-bit codes (-2048 to +2047) are linearly related to true voltage levels of -10 mV to +10 mV, respectively. Such linear functions are easily determined from the full-scale table, and are used to compute true voltages from A/D codes for cursor displays and exporting of numerical data. The 'Full-scale' column is automatically copied into the HyperCard global **FSTable** so you can access the full scale ranges for all A/D channels anywhere in the stack. The line from **FSTable** corresponding to the selected A/D channel can be directly copied to line 4 of the 'Hardware Parameters' field to save the current gain setting (see below). See the chapter on globals for a detailed explanation of how **FSTable** is used.

Similarly, the outputs of the D/A converters might be scaled down (or up). Enter these scale factors into the 'External D/A gain' table as you would for the external A/D gains. Note that if your D/A output is divided down by a factor of 10, you would enter 0.1 in the cell corresponding to that D/A channel. The resulting true full scale output of that D/A channel is shown in the 'D/A Range' column in volts.

Technical note:

As with the A/D converter, the gains for both D/A channels are available in HyperCard global **DACGainTable**. For the example in Fig. 6-8, line 1 of this global would be 1, and line 2 would be 0.1. Use this table to compute the correct parameters for passing to the XCMDs that generate analog pulses. Chapters on scripting and XCMDs/XFCNs give more details.

---

**Important Note:**

**Make sure to change the default A/D and D/A external gains to match your devices or you will get incorrect voltage readings.**

---

# *WaveTrak Cards*

All the cards discussed in this chapter are standard cards that WaveTrak needs in order to function properly. When you create a new, empty stack from an existing one, WaveTrak has to know which cards are standard cards, and which ones are your data cards. The 'Standard WaveTrak cards' table lists the *HyperCard* names of all standard cards. *Only the cards whose names are listed in this field will be duplicated to each new stack.* Therefore, don't delete any entries from this field or your new stack will not function. Also, if you create additional cards that will be permanent additions to all future stacks, you must give each of these cards a unique name and enter this name into this table (uncheck the Lock Text property of this field first). You can go to any card whose name appears in this table by clicking on its name in the field. This is how you can jump to infrequently visited cards which do not appear under the Go menu.

You will recognize the 'Hardware Parameters' table from the section on the trace card, above. In fact this table is used as a template whose contents are copied to each new trace card. You can edit both the legends (left hand side) and parameters (right hand side) according to your needs. We recommend that you do not change lines 1 through 4 as these are standard entries. In fact, this table is an alternate way of entering the number of points/wave and sampling interval (in µs) for subsequent acquisitions. This field is linked to the 'SAMPLING RATE' window in the Scope card; changes entered in one card are automatically reflected in the other. It is more convenient to program the sampling rate from the Scope card. Lines 5 to 8 are pre-defined to match standard WaveTrak XCMDs, but you can certainly rename these to match your needs. Use lines 9 and 10 for your own parameters.

The 'Readings' table also serves as a template for new Readings fields in trace cards. You can enter default values in the right column for readings that change infrequently, and change the legends in the left side to suit your application. The numbers to the left of each field represent line numbers to help you when you write your own scripts and want to fill in individual lines with data.

**Example 1**:

You set up experiments which will be performed at room temperature for the next several months. Click on the 37 entry beside temperature and enter 22 (room temperature in degrees Celsius, for example). This value will be copied to each new trace card as the default. It's more convenient to manually enter defaults that change infrequently into this table.

**Example 2**:

You are doing experiments where the temperature varies, and must be measured precisely and logged with each trace. Connect your temperature signal to an A/D channel. Add a few lines of HyperTalk script to your acquisition buttons to read this A/D channel and compute the temperature from the analog voltage. After each trace is created, the default of 37 is written to line 1 of its Readings table. Overwrite line 1 with your own temperature measurement.

Finally, the check box near the bottom right of the card, 'Hardware Lockout' allows you to disable all hardware access from startup. We recommend that you check this box after you are finished accumulating traces into a stack and the stack will only be used to review and analyze existing waves. There are two reasons for doing this. First, if you launch your data stack on another Mac to examine your data, and that machine has no acquisition card, you will get error dialogs each time you open the stack telling you that the hardware check failed and that you do not have a data acquisition card installed. By checking the Hardware Lockout box (its state will be remembered even after you quit HyperCard), hardware checks are bypassed and you won't get any error dialogs. Second, as hardware and system software change, it is possible, though unlikely, that certain low-level hardware operations performed by WaveTrak at startup could fail and force HyperCard to quit before you can get at your data. Checking Hardware Lockout will help to ensure that your stack will run on present and future CPUs and system software. If something goes wrong and you forgot to check the box, holding down the option key while you launch WaveTrak will have the same effect as the Hardware Lockout box. Keep the option key down until you get to the Home Card. This will permit you to access your stack and check the hardware lockout box.

## *Digital Filter Parameters Card*

WaveTrak contains a powerful set of signal processing functions, including the ability to digitally filter acquired or imported signals using the Fast Fourier Transform (FFT) or Finite Impulse Response (FIR) filters. For instance, you can remove noise from signals using filters that would be very difficult to implement with analog circuitry (such as very steep roll-off filters).

Technical note:

Digital filtering will *not* remove spurious signals created by aliasing because the filter has no way of knowing that aliased signals were not originally present at the source. Therefore make sure you choose a sampling rate fast enough, or filter your signal with an analog filter before sending it to the A/D converter.

The two types of digital filters presently supported includes FFT-based and FIR filters. Each has advantages and drawbacks. The 'Digital Filter Parameters' card (Fig. 6-9) allows you to directly enter filter parameters for all FFT filters presently supported by WaveTrak (version 1.4 supports hi-pass and lo-pass filters, with linear or log roll-offs). This allows you to modify the filter characteristics directly and on-the-fly. The disadvantages include longer computation times than (short) FIR filters and the requirement that waves contain a number of points that is an integral power of 2 (for the FFT algorithm). Furthermore, you have access only to the types of filters presently supported, as illustrated on the card. FIR filters on the other hand can operate on data sets of any length (not necessarily a power of 2), and are generally faster than FFT filters (provided the number of coefficients is small, e.g. 100 or less). The disadvantage of course, is that designing FIR filters is not trivial. We recommend the Igor Filter Design Laboratory in conjunction with Igor (both from WaveMetrics, Lake Oswego, OR). This package lets you design a wide variety of hi-, lo-, band-pass and arbitrary FIR filters. Copy the resulting coefficients from Igor and store them in the field in the FIR coefficients field by pressing the 'Paste Coeffs' button. The coefficients are copied into the field and automatically converted into a float wave in the global FIRcoeffs. Use this global when calling FIR filtering routines to determine the type of filter to be applied to your signal (see the 'FilterWaveFIR' XFCN in chapter 11, WaveTrak XCMDs and XFCNs, for more details). Type in a brief description of the FIR filter you just pasted into the 'Filter Description' field so you'll remember the characteristics. It is common to express digital filter frequencies using normalized frequency (0 to 1); this simply expresses the frequency as a fraction of the sampling frequency. For instance, if your sample rate is 10 µs per point (100 kHz), then a filter with a cutoff of 0.2 means 20 kHz. The filter frequencies for an FIR filter will therefore depend on the sampling frequency.

If all you need is low pass FIR filtering, version 1.4 now supports the design of low-pass FIR filters directly from within WaveTrak. An optimized convolution algorithm was added to exploit filter symmetry and speed the implementations. See the 'DesignFIRlo' and 'ConvolveWave' XFCNs in chapter 11.

The advantage of selecting parameters from a central location (as opposed to typing in values in a filter button, for instance) is that these values will be globally available to all filtering functions in the stack. When you decide to filter your data at different frequencies (when you start studying a different signal source for example), enter the new filter parameters in the Filter Card, and all filter buttons will immediately reflect the new changes; you won't have to update entries in many buttons each time you change your filter frequencies. All standard WaveTrak filtering buttons perform their filtering according to the global parameters entered in this card.

Although the card shows only lo-pass FFT filters, hi-pass FFT filters will use the same corresponding parameters, only their transfer functions will be mirror images of those shown.

The card does not perform error checking, so that a negative frequency, or fHi < fLo, will be accepted. However, this will result in an error when the filter XFCN is called. Note that all frequencies are in Hz, and the roll-off for the log filter must be positive (if you pass a negative roll-off value to the lo-pass filter, high frequencies will be *emphasized* instead of attenuated. This is not trapped as an error because such a filter can be useful under certain circumstances, such as sharpening up edges for easier detection. High frequency noise tends to be a problem with such filters however.) . Likewise, FIR coefficients are not checked so it is your responsibility to ensure their accuracy.

Technical note:

The parameters from the Digital Filter Card are copied to the following globals: **fLo**, **fHi** for linear roll-off FFT filters, and **f3dB**, **rolloff** for log roll-off FFT filters. The FIR coefficients field is converted form ASCII into a float wave and stored in the global variable **FIRCoeffs**. You can access the current filter parameters via these globals in your scripts.

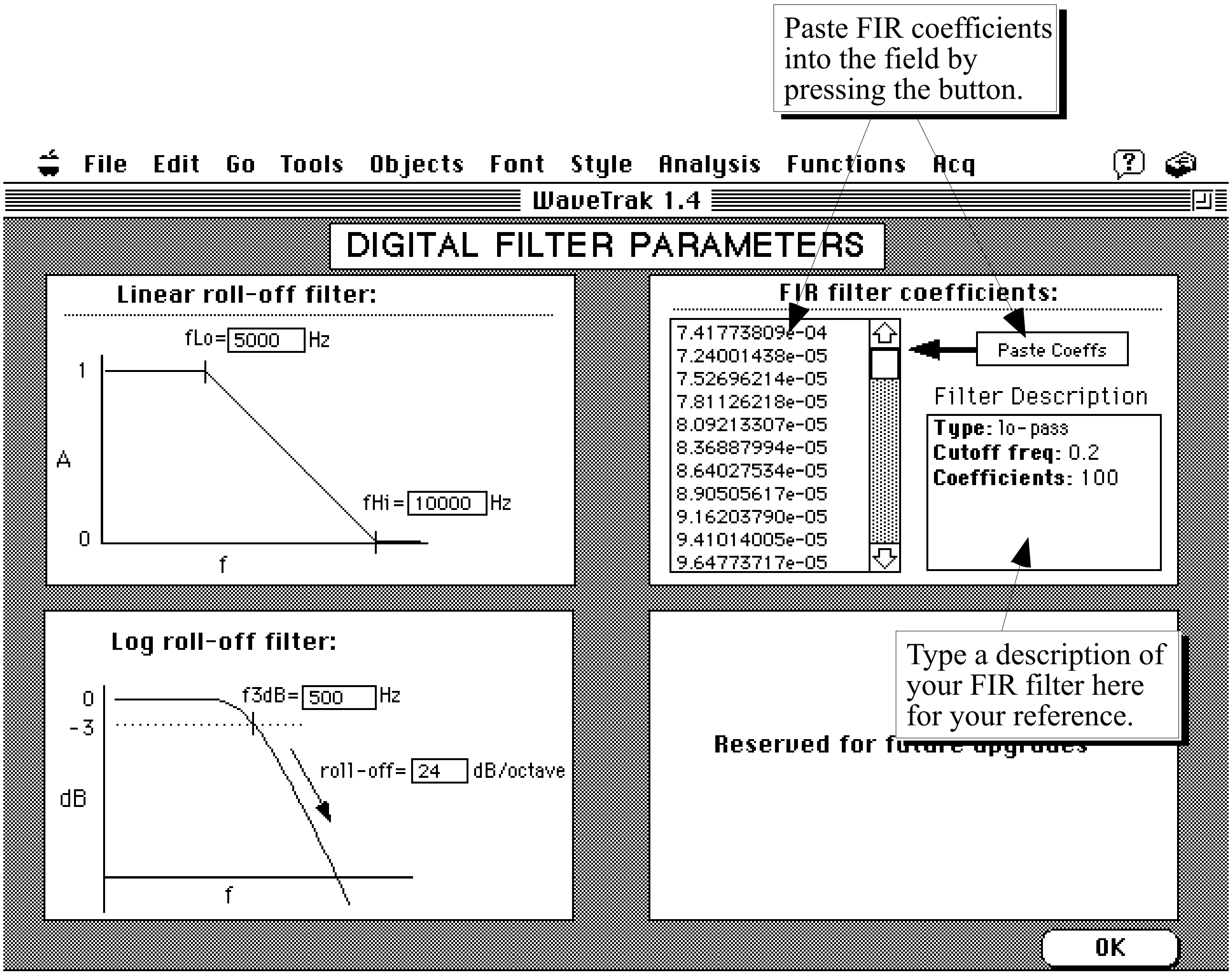



Fig. 6-9: The Digital Filter Parameters card allows you to enter FFT filter parameters directly, and to store the coefficients of an FIR filter designed using another software package. These values will be globally available to all filter functions in the stack. Although the drawings show only lo-pass FFT filters, hi-pass filters will use the same parameters, only their transfer functions will be mirror images of those shown here. The transfer function of the FIR filter will be determined by its coefficients.

## *Slot Address Card (WaveTrak AD version only)*

Unlike most other NuBus cards, the MacADIOS II card does not automatically inform the Mac of its existence or location at system startup. You have to manually tell WaveTrak where you installed the card. The 'Slot Address Card', shown in Fig. 6-10, is a map of the NuBus slots present in the Macintosh II, IIx, IIcx, IIci and IIfx models.  When you

click on a slot, WaveTrak will check for the presence of a MacADIOS II card and inform you if all is well. An error dialog will inform you if the slot is empty, or contains another type of card; the chapter on WaveTrak error messages will help you figure out why WaveTrak doesn't agree with your selection. You only have to do this once; WaveTrak will automatically look in this card for the selected slot at startup. *Remember to update the correct slot if you change the location of your card or upgrade to a new Mac.*

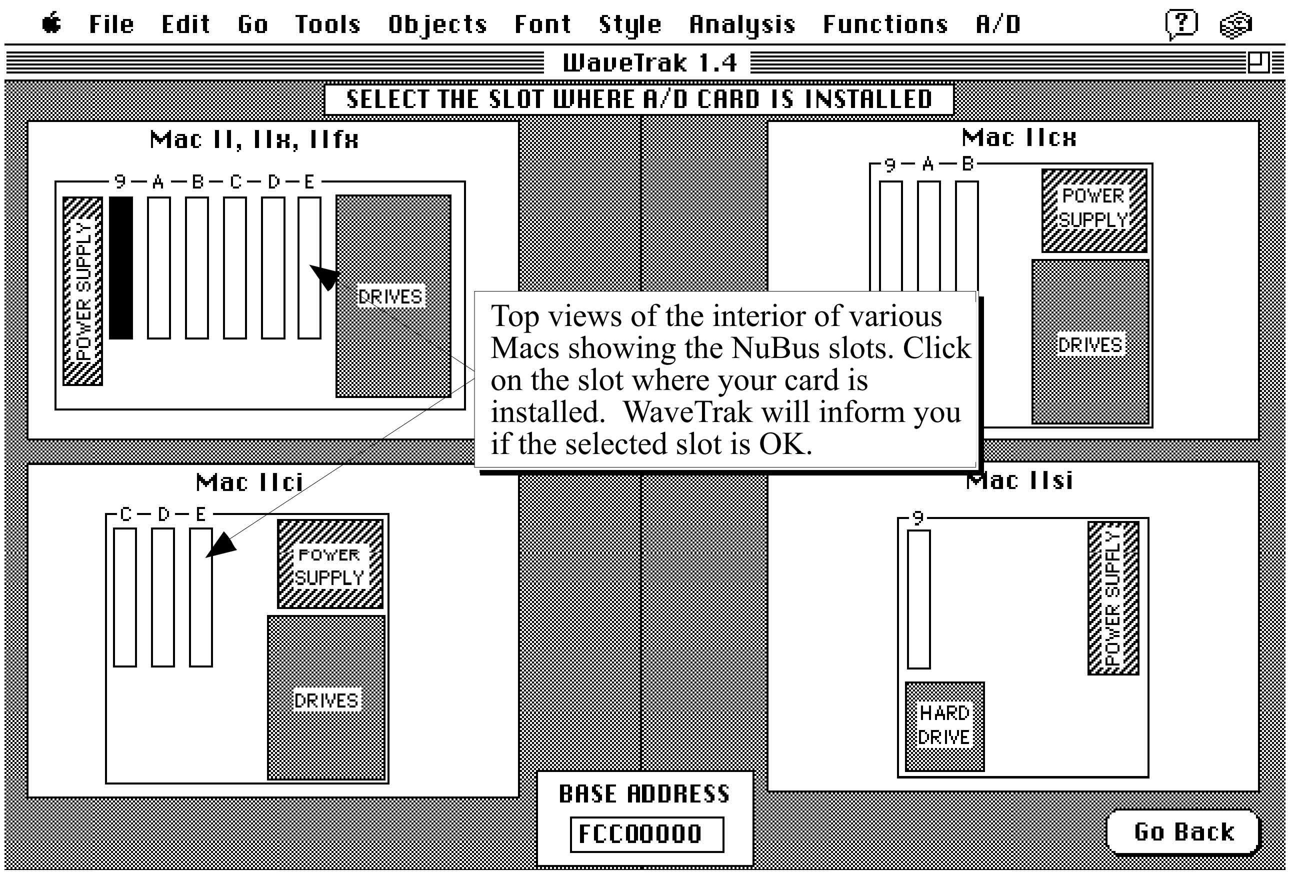



Fig. 6-10: the Slot Address card contains illustrations of the interiors of various Macintosh models. Select the slot where your card is installed by clicking on it.

Technical note:

Macintosh NuBus slots are mapped onto the 68020/030 address space. Each slot has a certain amount of address space allocated to it. The 'BASE ADDRESS' corresponds to the first address of this block where a device on the MacADIOS II card can be addressed. WaveTrak finds various devices on the MacADIOS II card by adding a device-specific offset (which never changes) to the base address which you provide from the Slot Address card.

Note: for newer Macintosh models, use the generic Mac II slot buttons to select the slot where you think your card is mapped to. All Macs have there slot base addresses at F9900000, FAA00000, FBB00000, FCC00000, FDD00000, FEE00000. The automatic hardware check will inform you when it finds your MacADIOS II card in one of the slots.

## *Button Bank*

The Button Bank card does not perform any operations, but merely serves as a repository for a variety of pre-programmed buttons. You can jump to the Button Bank via the Go menu. You will normally go to this card to copy a button (don't *cut* it or you will lose the original), then paste it wherever you need. The function of each button is described in the WaveTrak Buttons chapter and in the form of comments within the button script. The ACQUISITION, MEASUREMENT and SIGNAL GENERATION groups require a MacADIOS II A/D board and the *A/D version* of WaveTrak. The ANALYSIS & WAVE MATH group operates on either digitized or imported waves. When you create your own buttons that you want to save for later, deposit them in the Button Bank for later retrieval. This card, along with all its buttons, will be duplicated in all new stacks, so all your buttons will move with you.

Buttons in the button bank usually require that they be pressed from within a root or trace card. *Don't press any buttons from within the Button Bank card or you will get an error*. The message box reminds you each time you go to this card.

## *NOTES Card*

The NOTES card is a general purpose card with a single scrolling field where you can type any notes or reminders. This card, along with its contents, will be duplicated in all new stacks. You can jump to the NOTES card from the Go menu.

*Miscellaneous Cards*

There are several other cards with which you will never interact directly. Briefly, the ErrorList card contains error codes and brief explanations for each error generated by WaveTrak XCMDs/XFCNs. The handler **ErrNum** in the stack script automatically displays the message when you pass it the error number (always returned by XCMDs in global **XCMDErr**). The 'Plot card' is a blank card used for drawing graphs and other graphical data. Finally, the 'StdPostedCard' is used to clone Posted-cards. See the WaveTrak Menus chapter for details.

**Table of standard WaveTrak cards and their HyperCard names:**

| **WaveTrak Card** | **HyperCard Name** |
|---|---|
| Home Card | HomeCard |
| Standard Root Card | StdRootCard |
| Standard Root Comment Card | StdRootComment |
| Standard Trace Card | StdTraceCard |
| Scope Card | Scope |
| System Parameters Card | SysParams |
| Slot Address Card | SlotAddressMap |
| Button Bank | ButtonBank |
| Notes Card | Notes |
| Plot Card | PlotCard |
| Standard Posted Card | StdPostedCard |
| Error List Card | ErrorList |
| Digital Filter Parameters Card | FilterCard |

## In Summary

- WaveTrak consists of a number of standard cards which perform the housekeeping.
- Your data is stored on root and trace cards.
- One root card represents the first card of each new experiment; it contains summary data and comments pertaining to the entire experiment.
- Under the root card are trace cards. Each trace card holds one wave along with other information and readings taken at the time, thus completely describing the acquisition.
- You navigate among roots and traces using arrow buttons.
- Standard cards are accessed through the 'Go' menu.