TF32 User's Manual

Paul H. Milenkovic [1]
Department of Electrical and Computer Engineering
University of Wisconsin-Madison
1415 Johnson Drive
Madison, Wisconsin 53706

[1]Please refer inquiries to Paul Milenkovic, 118 Shiloh Dr., Madison, Wisconsin 53705-2433, U.S.A., 608/833-7956, `http://userpages.chorus.net/cspeech`, `cspeech@chorus.net`

Abstract

TF32, time-frequency analysis software program for 32-bit Windows, displays and analyses speech or other audio-frequency waveforms and is able to

Play up to two audio channels using any Windows sound card. Play a sequence of selected intervals from one or more waveform files with controllable pauses.

Record up to two audio channels from the Windows sound card (TF32 Basic level or higher). Record up to 16 DC-coupled channels at differential rates using an A/D card compatible with the Data Translation Open Layers Windows driver (requires TF32 Lab Automation level).

Open waveforms in any of these file formats: CMU (pre-release TIMIT), old CSpeech format, CSRE 4.0, ILS, Kay Elemetrics (.NSP), NCVS92 (new CSpeech format – normal or lossless compressed), RIFF (Microsoft .WAV), Speech Station, SPHERE (current TIMIT).

Export waveforms in RIFF, SPHERE, or lossless-compressed formats;

Compute pitch, RMS, and dB traces, inverse filter voice source waveform.

Display up to two time-frequency spectrograms with editable LPC formant overlay.

Mark and display waveform text labels.

Display Fourier, LPC, or moments time-slice spectrogram.

Measure waveform jitter, shimmer, and voice periodicity SNR;

Save, retrieve, and edit pitch and formant values in ASCII-format files.

Copy wave and analysis plots, spectrograms, and numeric readouts into documents using the Windows Clipboard.

Automatically analyse a sequence of selected intervals from one or more waveform files, saving the numeric values to an ASCII file.

While the commands for controlling the display follow the CSpeech software program for DOS closely, TF32 is a completely redesigned program to take advantage of Microsoft Windows. Analysis displays update automatically when you open, record, or filter a waveform; numeric readouts change you move the waveform cursors. The increase from 16 to 128 grey-scale spectrogram levels gives better rendering of detail. The waveform plots, along with real-time updates of the spectrogram, pitch trace, or other analyses, scroll during recording, playback, or in response to selecting a scroll bar control with the mouse. When permitted by your graphics card and monitor, screen updates are synchronized with the video refresh cycle to give smooth motion of waveform and spectrogram displays comparable to a dedicated hardware instrument.

Forward

The name selected for a product during its development becomes widely known and it sticks. TF32 was named because a grey-scale time-frequency spectrogram was the first feature implemented and 32-bit Windows (95 and NT 4) played a role in both spurring and enabling its development. By now, I am unable to call it anything else.

I am told there is a Russian saying "Better is the enemy of good enough." I did not switch from the DOS CSpeech to the Windows TF32 willingly. CSpeech and DOS remain "good enough" for many speech analysis tasks, and switching existing acoustic analysis algorithms over to Windows took time away from developing new algorithms. By 1995, however, the release of Windows 95 and NT 4 offered a carrot in the form of a 32-bit address space, greatly simplifying the data structures internal to the software for managing waveforms longer than a few seconds in duration. The stick was that Windows NT 4 no longer allowed CSpeech running in a DOS window to access hardware registers required to record and play waveforms. I have been telling CSpeech users to run Windows 98 and to purchase genuine Creative Labs sound cards because they are compatible with the Sound Blaster 16. Now I find Creative Labs is selling sound cards no longer compatible with the Sound Blaster 16 and hence unusable from the DOS window under Windows 98. The processes is like watching a slowly rising draw bridge uncovering an unpassable moat.

Being a late adopter of Windows, I decided to take the time to design TF32 to work as effectively with Windows as possible rather than make a quick translation of CSpeech. The relentless march of Moore's law meant that I had much more processor power available to develop more highly interactive displays than when I designed the interface to CSpeech. TF32 employs analysis displays which update automatically when you open, filter, or record a waveform; CSpeech requires the user to explicitly enter commands to update displays.

I have been fortunate that I have been able to reimplement and also improve what I have written into CSpeech in the form of TF32, and I would like to thank the ongoing forebearance of my faculty colleagues of the ECE Deparment at the University of Wisconsin-Madison as well as of speech science colleagues waiting for TF32 to emerge. In addition to acknowledging the role of Charles Read, I would like to acknowledge the contributions of many persons outside the ECE department starting in the CSpeech days: Cliff Gillman, David Wilson, Rick Konopacki, William Henke, Pat Keating, Jerry Jacobsen, Diane Bless, David Drucker, Christopher Dromey, Gary Weismer, Ray Kent, Karen Forrest, Susan Nittrouer, Bruce Poburka, Eugene Buder, James Dembowski, John Westbury, Stephen Tasko, Marilyn Workinger, Lisa Bast, Jacqueline Laures, Yana Yunosova, and others who helped shape CSpeech into its present form.

Paul Milenkovic, Madison, Wisconsin.

Contents

List of Tables

1 Introduction

TF32 is a software program for visualizing properties of the acoustic speech signal or other audio-frequency waveforms by displaying waveform plots along with pitch, spectrograms and other analyses computed from those waveforms.

You can display waveforms by recording or by retrieving waveforms from files by selecting from the `Files` menu. For any given set of displayed waveforms, you may add analyses to the display, modify their parameter settings, or remove analyses by selecting from the `View` menu. When the displayed waveforms change in response to recording, opening a new waveform file, or filtering a waveform, the analyses are automatically recomputed.

TF32 is designed so that waveform recording and playback, opening a file, or filtering to a waveform does not impede making selections from the menu. There are no "hourglass" wait cursors in TF32 for performing any of those actions, and the updating waveform and analysis displays are their own "progress meter" when opening extremely long files. Analysis displays may be added, changed, or closed while the display is being updated, or a long display update may be cancelled at any time by opening a new waveform or by exiting the program.

During recording, waveform playback, or in response to selecting a scroll button with the mouse, waveforms plots along with time-aligned analysis plots scroll across the screen, making TF32 a real-time pitch, intensity, or time-frequency spectrogram analyser. When permitted by your graphics card and monitor, screen updates are synchronized with the video refresh cycle to give smooth scrolling of waveform and spectrogram displays comparable to a dedicated hardware instrument.

TF32 uses two ways of putting displays into your documents. Both ways use features of Windows and do not require the use of special software utilities. Pressing the `Alt-Print Scrn` key combination copies a bit-map image of the active TF32 window into the Windows clipboard. Use this way if you want an image of the entire TF32 display including menus. The `Copy image to clipboard` command selected from the TF32 `Edit` menu copies a Windows metafile image of the waveform plots and analysis displays into the Windows clipboard. Use this way to generate figures in your document showing detailed waveform plots and analyses without including the TF32 menus and window frames.

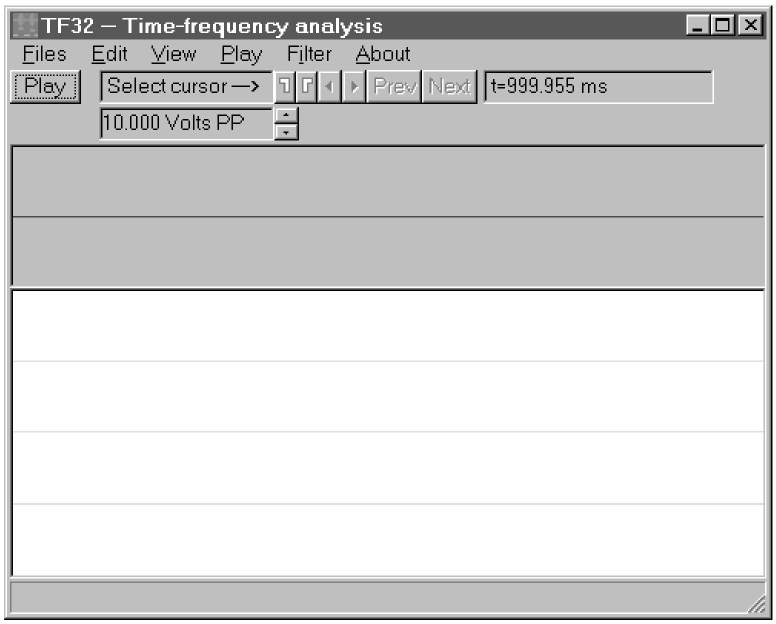
Numeric readouts appearing in waveform or analysis displays may be copied as text to the clipboard and pasted into text files or into spreadsheet forms. The use of tabs to separate descriptive text from numbers helps export data to spreadsheets. TF32 also has a "batch mode" for making measurements. You can specify a list of selected intervals of one or more waveform files and automatically sequence through that list and save numeric readouts to an ASCII file.

2 Starting TF32

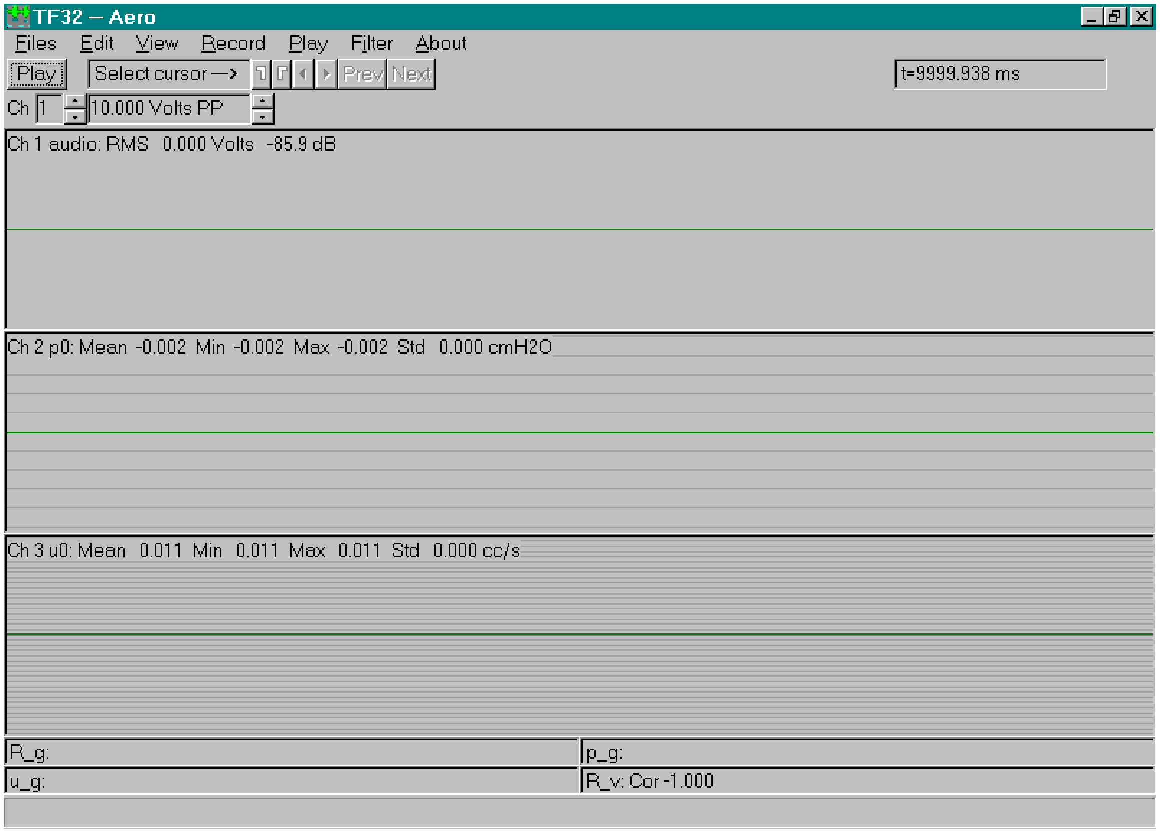
Invoke TF32 by entering

```
tf32
```

at a DOS window command prompt or by double-clicking the TF32 icon on the Windows desktop display – check with the person who installed TF32 on your computer on how to invoke TF32. You will get the display

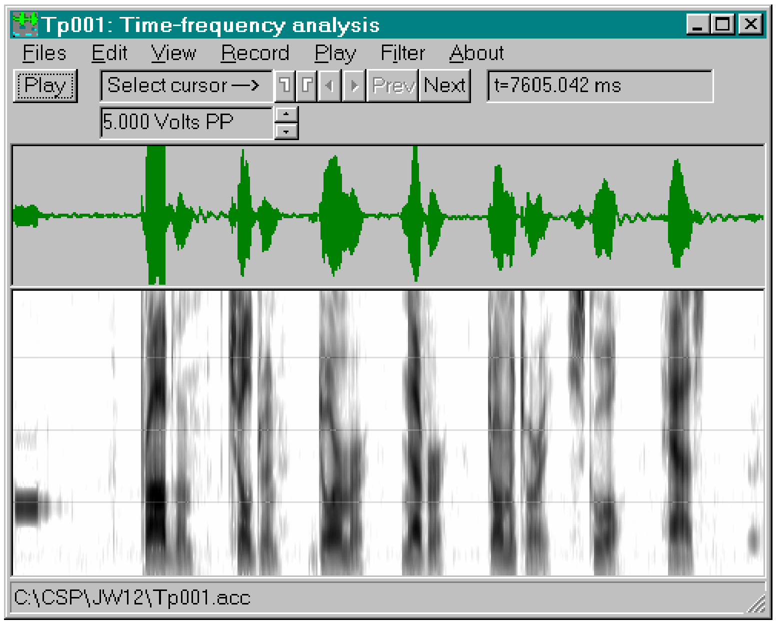


If TF32 has been installed to display a preset set of waveform plots, you may get the display



This display is the preset configuration of an audio channel, a pressure channel, and a flow channel for voice aerodymanic measurement.

Going back to the first case, opening a single-channel waveform file with the `Files Open` command results in the display



The first line of the TF32 main window is the title bar; it indicates that the waveform from file called `tp001` is plotted in the display. The last line is the status line; it shows the complete path name for this file. The main window menu appears below the title bar, and below that are a set of controls and indicators. These controls and indicators specify the amplitude scale and time base of one or more waveform plots that appear below the controls.

The main window display shows a single waveform plot and below that a grey-scale time-frequency display computed from that waveform. If the waveform is filtered with a selection from the `Filter` menu or if it is replaced with a different waveform by invoking `Files Open` on a different file, the time-frequency display will be automatically recomputed to show the spectrogram of the new waveform.

3 Menu commands

The main menu selections are

Files specifies the storage and retrieval of waveforms from files. Exit TF32 by selecting `Exit` from this menu, by clicking on the X on the upper right hand title bar, or by clicking on the icon on the title bar and selecting `Close`.

Edit specifies how waveform plot and analysis images as well as numeric values of waveform measurements get exported into other software packages. Images are copied to the Window clipboard for insertion into documents (invoke the Paste command of your word processor); numeric values are copied to the clipboard or are appended to text files.

View specifies the display of numeric readouts of waveform measurement along with the display of spectrograms and analysis traces computed from the waveforms. While the `Files` menu selects waveform data, the `View` menu specifies how those waveforms are viewed.

Record displays a dialog for specifying how waveforms are acquired from the A/D.

Play offers options for waveform playback through the sound card.

Filter lists filters that can be applied to the waveform plots.

About displays the version and revision of TF32.

To summarize, the `Files` menu saves and retrieves waveforms, `Edit` exports display images and numeric readouts, `View` controls how the waveforms are viewed and analysed, and `Filter` applies transformation to waveforms; the analysis displays are automatically updated when a filter is applied to an analysed waveform.

A description of all the menu commands follows.

3.1 Files menu

The `Files` menu lists commands to 1) retrieve waveform files for display, 2) save waveforms into files, 3) select waveforms for display from a sequence of related waveforms, and 4) exit the TF32 program.

The `Files` menu commands for retrieving waveforms are

`Open` inputs any of four file types recognized by TF32: 1) a waveform file containing one or more signal waveforms, 2) a wave list (text file containing a list of waveform files selectable with the `Prev` and `Next` commands), 3) a text file containing a waveform header describing the layout of waveforms stored in headerless binary format, or 4) a preset file (text file containing settings for displays and for data acquisition – requires Lab Automation level).

`Add plot` displays another waveform file by adding one or more waveform plots to the display. Allows displaying waveforms from more than one waveform file.

`New` plots a zero-valued waveform display – available channel selections are the same as the `Record` command.

TF32 opens waveform files of these formats: CSpeech, NCVS92 (new CSpeech format), RIFF (Windows .WAV – 8 or 16 bit), UW Microbeam Database compressed (.ACC), TIMIT Database (CMU and SPHERE) formats, Kay Elemetrics .NSP, ILS, Speech Station. TF32 can also open a waveform stored in headerless binary format. If such a file has extension *.ext*, first open a header file named *ext*`.hdr` to enable opening *.ext* files as waveforms. The text-format header file supplies the necessary format information – number of channels, sampling rate – to open a headerless file.

The `Files` menu commands for saving waveforms into files are

`New session` specifies subject/patient (keyed by id number) and recording session (keyed by date) for automatically saving waveforms acquired with `Record` and for automatically logging measurements of those waveforms.

`Save recorded` explicitly saves a waveform acquired with `Record`. If settings in the preset file specify waveform calibration information – DC offset and calibrated signal range – the file format selections will be restricted to formats that preserve that waveform calibration information.

`Export wave` saves a waveform retrieved with `Files Open`. This command is not restricted in its selection of file formats, so calibration information may be lost in the exported file.

`Export hdr` saves an ASCII waveform header file to allow opening a waveform file saved in headerless binary format.

Both `Save recorded` and `Export wave` allow saving a cursor-selected portion of the waveform: between cursors, from left cursor to end of buffer, or from beginning of buffer to right cursor, depending on whether the left cursor or right cursor have been placed on the display. These commands save the entire waveform buffer when no cursors are placed on the screen, even if the waveform display is zoomed to display only a portion of the waveform buffer.

The `Files` menu commands specifying selection from a sequence of related waveforms are

- `Prev` opens the previous waveform from a sequence – same as the `Prev` button on the TF32 main window.
- `Next` opens the next waveform from a sequence – same as the `Next` button on the TF32 main window.
- `Select` allows random access to a sequence of waveforms.
- `Auto filt` selects/deselects the auto-filt mode. With auto-filt mode enabled, waveform filters are automatically reapplied when you select another waveform with `Prev`, `Next`, or `Select`. With auto-filt disabled (the default condition), you need to explicitly reapply any filters by making selections from the `Filter` menu every time you select another waveform.

When you input a waveform file with `Files Open`, you are implicitly specifying a sequence of related waveforms made up of all the files with the same extension in the same directory. The sequence is in alphanumeric order (`rec001` followed by `rec002`). The `Prev` and `Next` commands select waveforms in sequence order while the `Select` command allows random access to a waveform in that sequence.

If you open a wave list, TF32 displays the first waveform on that list, and the `Prev` and `Next` buttons allow selecting waveforms from the list in list order. You can create or append to a wave list by selecting `Add indices to wave list` from the `Edit` menu.

When you `Open` a wave file with one extension and `Add plot` one or more wave files with the same file name but different extensions, the sequence-selection commands `Prev`, `Next`, and `Select` maintain that grouping of files. For example, you can `Open tp001.acc` and `Add plot tp001.tcc` to display both the `.acc` and `.tcc` waveforms. Invoking `Next` opens `tp002.acc` along with `tp003.tcc`. This grouping of related waveforms is reset by `Files Open`, `Files New plot`, or `Record`.

3.2 Edit menu

The `Edit` menu lists commands to 1) copy displayed images or numeric measurements to the Windows clipboard or printer, 2) log measurements during a recording session, 3) make measurements of a sequence of waveform files, and 4) add the cursor-selected times along with the file name of the displayed waveform to a list file for making a measurement of such a sequence.

The `Edit` menu commands specifying the clipboard or printer are

`Copy image to clipboard` copies the screen display as an image to the clipboard while adding a time axis marked in 100 ms intervals. This image can contain waveform plots, the grey-scale spectrogram, pitch and RMS analysis traces, and text labels. The time-slice spectrum (`View Open Spec`) and the x-y articulatory display (`View Open XY`) have separate `Copy` commands to transfer those images to the clipboard. Invoke the `Edit Paste` command in a word processing program to place an image in a document for printing.

`Copy readouts to clipboard` saves numeric readouts on the display to the clipboard in text format. This includes numeric readouts from waveform plots, analysis traces, and analysis readout boxes. It also includes the time-slice spectrum readouts (`View Open Spec`) when the main window is maximized, which docks the spectrum window to that display. The analyses that produce readouts along with their readout settings are selectable from the `View` menu. Invoke the `Edit Paste` command to place as text in a document or spreadsheet – the fields are tab delimited to facilitate use of a spreadsheet.

`Copy image to printer` copies the displayed image and accompanying numeric readouts to the default printer – use the Windows printer manager on the control panel to select the default printer and set its resolution.

3.2.1 Batch-mode measurement of lists of waveforms

The batch-mode measurement of lists of waveforms is performed using the `Edit` menu commands

`Measure sequence` saves numeric readouts from a sequence of waveforms to the specified text file. This command will append a measurement to that text file, automatically invoke the `Next` command to select the next waveform in the sequence, append the next measurement to a file, and continue the cycle until it reaches the end of the sequence.

`Add index to wave list` appends the cursor selected start and stop times along with the file name of the displayed waveform to a text file.

Batch-mode measurement is a three step process. The first step is to create a list of waveforms to measure and to Open the first waveform in that list. The second step is to configure the TF32 display with the desired analysis displays and to select numeric readouts for the displayed waveforms and associated analysis displays – this is done by making selections from the `View` menu. Be sure and apply any required filters (high-pass filtering of the acoustic signal removes DC-bias from the RMS dB readout) – any filters you apply will be automatically reapplied to each waveform in the list. The third step is to invoke the `Edit Measure sequence` command described above.

What you see is what you get – you will see TF32 stepping through all the waveforms in the list and each of the visible numeric readouts will be added as columns to a text file, with a new row for each waveform in the list. If you interrupt `Measure sequence` by pressing the `Stop` button, the waveform in the display will not have been measured. That waveform, however, will be the first one measured if you reinvoke `Measure sequence` to resume the automated measurement.

There are two ways to specify a waveform list. The first way is to open one of several files in the same directory with the same file extension. The list will be that file along with all the files with the same extension that follow in alphabetical-sort order. This way only allows measuring the entire file instead of a selected portion of a file.

The second way to specify a list is to use the `Edit Add index to wave list` command described above to create a text file with the `.lst` extension containing file names along with cursor-selected time intervals. The same file name can appear multiple times with different cursor-selected times if you want to analyse multiple intervals of the same waveform. This way will measure the selected portion of each file on the list. Opening the list file with the `Files Open` command will open the first file on that list and position the waveform display to the selected time interval. Stepping through the list will select each listed file name and time interval in the order they appear in the list file.

3.2.2 Logging measurements during a recording session

The Edit menu commands for logging measurements during a recording session (requires Lab Automation level) are

Annotate session log enters a text line into the session log file in the directory specified by Files New session.

Log readouts/Append readouts to log enters numeric readouts for a waveform acquired with the Record command into the session log file. Log readouts logs an initial measurement of a recorded waveform; Append readouts to log means you are logging additional measurements of that waveform that may require different cursor placements or analysis selections.

Update and view log/View log opens the session log file with the Windows Notepad program, allowing you to review entries, add additional notes, or copy the contents of the log to the printer. You need to close Notepad to resume operation of TF32. Update and view log means you are logging the initial measurement of the recorded waveform before viewing the log with Notepad; View log means that you have already logged the initial measurement for that waveform with Log readouts.

To log measurements, select New session from the Files menu to specify a directory for the waveform files and session log for that recording session. If TF32 had been installed to start in the directory c:\aero, if a subject named Joanna Q. Subject with ID number 123456789 has been entered into the New session name and ID fields, and if the session date is March 20, 2001, the session directory is

c:\aero\data\Subject.JW.123456789\20Mar2001.

The session log file within that directory is called

Subject.JW.123456789.20Mar2001.txt –

this long file identifying the subject, the subject ID, and the session date will appear as the header on printer output produced by Notepad selected from TF32 with the Edit Update and view log command.

After invoking Files New session, selecting Record from the main menu starts a waveform acquisition and automatic measurement logging cycle. When acquisition is done,the recorded waveform is saved in the session directory and the Filter Preset (if specified in the preset file) is automatically applied to the waveform plots. The record name (selected from the sequence rec001, rec002, and so on) is automatically appended to the log. If you select Annotate session log from the Edit menu at this point, you have an opportunity to add a remark or note that immediately follows the record name.

You now have the opportunity to change the displays and readouts with the `View` menu or to set cursors to select a time interval for measurement. You can explicitly log an initial measurement of the recorded waveform by selecting `Log readouts`, or you can let TF32 automatically log the initial measurement by viewing the current log file with `Update and view log`, acquiring another waveform with `Record`, examining a previously-recorded waveform with `Files Open`, or by exiting TF32. That way at least one measurement is logged for every recorded waveform.

If you log an initial measurement with `Log readouts` (or `Update and view log`), you can log additional measurements for the same recorded waveform with `Append to log`, making multiple measurements of the same waveform with different cursor placements or analysis selections.

The action of logging measurements during a recording session generates a list file automatically. Specify a recording session with `Files New session`, acquire waveforms with `Record`, log measurements with `Edit Log measure` or `Edit Update and view log` – these actions generates a list file called `measure.lst` in the session directory. While acquisition logging takes place during a recording session, if you open `measure.lst`, you can conduct post processing on that sequence of waveform files and cursor positions with the `Edit Measure sequence` command.

If acquisition logging applied a preset filter (typically a high-pass filter of the audio channel to remove DC bias from the RMS dB readout), be sure and reapply that filter with the `Filter Preset` command after opening `measure.lst` for post processing. The reason acquisition logging saves raw, unfiltered waveforms to archival files is to give you a chance to either view the raw data or to apply a different filter in post processing – once a stored waveform is filtered you cannot get the raw waveform back.

3.3 View menu

The `View` menu allows 1) adding new analyses to the display, 2) modifying settings of analyses already on the display, and 3) removing analyses from the display.

Selecting `View` from the main menu drops down a menu containing an entry called `Open` followed by entries that name displays that are already viewable, listed in the order they appear on the TF32 main window. Selecting `Open` activates a submenu that in turn lists analysis displays that can be added to the main window. When you open a display in this way, it is added to the list of viewable displays.

TF32 for example is limited to two `TimeFreq` displays. When you open your limit of `TimeFreq` displays, `TimeFreqA` and `TimeFreqB` appear in the list of displays, but the entry to add another `TimeFreq` display is removed from the `Open` submenu. When you close one of the `TimeFreq` displays (select it from the list of displays on the `View` menu, check `Close TF` and select `OK` in the time-frequency dialog), the entry to add it back reappears in the `Open` submenu.

To change or remove a display, select that display from the `View` menu. If the menu entry appears without a check mark, selecting that entry brings up a dialog for changing settings or removing that display. If the menu entry has a check mark, selecting that entry will remove the display. Changing settings for that kind of display requires selecting controls on the display with the mouse; the time-slice spectrum display and the xy articulatory displays have such direct controls not requiring a dialog.

The waveform plots are configurable with the `Wave plots` entry in the `View` menu, which brings up a dialog to show/hide waveforms, set horizontal grid lines on waveform plots and select numeric readouts for different sets of waveform measurements. As the waveform plots are controlled from the `Files` menu, this dialog only hides waveforms (useful if you want more of the screen to show detail in an analysis display); it does not add or remove the underlying waveform plots.

The analysis displays all refer to at least one waveform channel and are all removable. They are manually removable from the `View` menu; they are removed automatically when you update the waveform plots with `Files Open`, `Files New`, or `Record`, and a waveform referenced by that analysis is not longer plotted. For example, if a time-frequency spectrogram is viewing Ch 2 and then you open a single-channel waveform file, that time-frequency display will be closed. The text label and xy-articulatory displays only refer to Ch 1; as Ch 1 always remains after `Files Open`, `Files New`, or `Record`, these displays are not automatically removed.

What follows are standard analysis selections, depending on the version of TF32. In the Lab Automation level of TF32, the preset file can specify "canned" sets of analysis displays – these appear as added entries on the `View Open` submenu.

The analysis displays selectable from `View Open` include

`Pitch trace` is pitch-period adaptive cross-correlation pitch analysis using LPC inverse filtering to suppress false identification of low frequency formant oscillations as pitch cycles. The pitch trace changes on the identified voice excitation boundaries. Computed pitch traces are plotted in olive green, pitch traces input from a file are plotted in blue-green (cyan).

`RMS trace` computes the RMS trace of a waveform; linear and dB scale is selectable.

`uglot trace` computes the glottal flow waveform; flow or microphone source, flow or flow derivative are selectable.

`TimeFreq` has a dialog to select BW (45 Hz for narrowband, 300 Hz for wideband, 450 Hz wideband for high F_0 speech), preemphasis , frequency range, dB floor, dB range, and waveform channel for a gray-scale time-frequency spectrogram display. This dialog has additional controls to select an LPC formant track overlay and to edit those formant tracks.

`Label` display text labels. If the label file you select does not yet exist, you will see a blank label display; `View Mark label` will create this file when you mark the first label.

`XY` (Demo level only) displays the x-y articulatory plot (requires x-ray microbeam or other articulatory data in the .xyd file). Invoke the `Play` command from the TF32 main menu to play sound and animate this display.

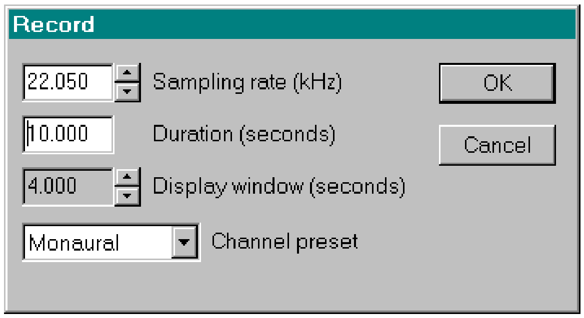
`Spec` Displays the time-slice spectrum of the cursor-selected waveform up to a maximum of 1024 samples; check the `Align wnd` box to center the analysis interval on either the left or right waveform cursor. Options are Fourier Hamming and rectangular window, LPC, discrete-cosine transformation smoothing, and moments model. The `M` button displays the current spectrum in gray for comparison with a new spectrum. The `Copy` button places the time-slice spectrum in the clipboard to paste into word processor or other documents.

`Time` (Demo level only) Display the time to compute the time-frequency spectrogram. Useful for comparing processor speeds.

`Jitter` computes the voice periodicity measures jitter, shimmer, and periodicity SNR from the cursor-selected interval of the selected waveform channel.

3.4 Record dialog

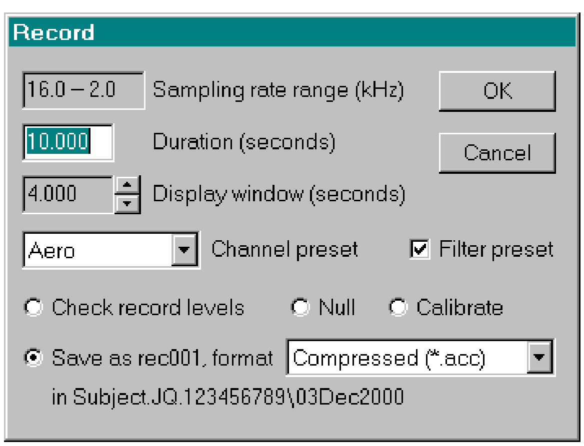
The `Record` command on the TF32 main menu activates the Record dialog.



This dialog allows selecting the sampling rate, buffer duration, length of the display window for the scrolling waveform and analysis display during recording, and the channel preset. The channel presets `Monaural` and `Stereo` select recording on one or two channels from the Windows Multimedia sound card.

The preset file (requires Lab Automation level) can specify additional channel presets for sampling from a Data Translation A/D card. The sampling rate is fixed for those channel presets specified in the preset file. Depending on how TF32 was installed on your computer, activating TF32 by selecting an icon on the Windows desktop can activate TF32 and automatically read in a preset file to configure TF32 for a particular task.

If acquisition from a Data Translation A/D card has been specified in a preset file, and if automatic waveform logging has been enabled by entering a subject identifier into the dialog box of the `Files New session`, the `Record` dialog will look like



The `Files New session` dialog was used in this last example to designate a session for a subject with initials JQ, last name Subject, and id 123456789. The recording session took place on Dec 3, 2000. According to the data acquisition sequence, the waveform recording will be saved in a file named rec001.acc under the session subdirectory Subject.JQ.123456789\03Dec2000 under the subdirectory data under the startup directory from which TF32 was activated. The next waveform recordings gets saved in the file rec002.acc.

The Record dialog whill show a list box that allows selecting the file format for automatically saving the waveform recording. The format choices will be restricted to those preserving calibration information if calibration is enabled for the selected channel preset. If the selected channel preset requires differential sampling rates or if the Compressed waveform format is selected when saving multiple waveform channels, waveforms will be stored in multiple files. If the first channel is saved in rec001.acc, subsequent channels will be save in rec002.acc.2, rec002.acc.3 and so on. It is only necessary to open the file rec001.acc when retrieving the data; TF32 will automatically input the channels stored in the other files into the display.

The preset file may specify a filter preset to be applied to the waveform plots in the display to condition the recorded waveforms for measurement and data logging. Since the filter preset is applied to the waveforms *after* they are automatically saved, the waveform files contain raw signals before filtering. When you apply Files Open to one of these waveforms, you can optionally select the Filter Preset command; this applies the same signal conditioning filters used in the automatic logging of waveform recording. Or if you do not chose to select Filter Preset, you can view or measure the raw waveform data or apply different signal conditioning filters during post processing.

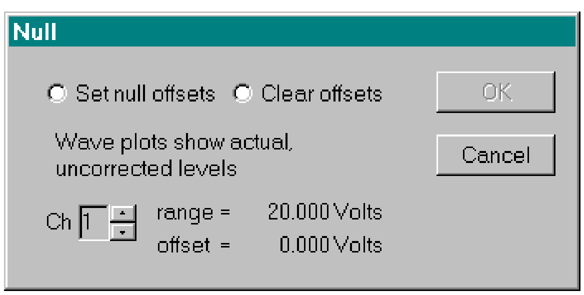
The Check record levels, Null, and Calibrate settings allow recording without automatically saving to a file and logging measurements. The Null and Calibrate settings are only enabled for channel presets for which the calibration feature is enabled in the preset file. When recording with one of these three selections, the filter preset will not be automatically applied, but it can be manually selected with the Filter Preset command. The Null and Calibrate selections will automatically bring up dialogs for the Null and Calibrate operations when recording completes or when recording is stopped by pressing the keyboard space bar or selecting the Stop button with the mouse.

3.4.1 Null and calibrate

The calibration of a waveform measurement requires two actions. Null insures that the waveform readouts show zero when zero-level inputs are applied to the transducers connected to the data acquisition channels. Calibrate measures a known value differing from zero that is supplied to a signal transducer; entering that known value into the TF32 program specifies the correct scale factor so that the waveform readouts show calibrated values.

Setting the signal null in software needs to be preceded by setting the channel balance; this means adjusting the offset control on the DC amplifier feeding that channel. This offset control is used to make a coarse correction to the amplifier zero-level reading. It is important to make this coarse correction by adjusting the offset control on the amplifier to insure that the amplifier zero-level is centered in the input voltage range of the A/D; if this is not done, the usable signal range will be greatly reduced. Once the coarse correction is made with the amplifier hardware, a fine correction may be safely applied in software.

Setting the balance and null of all input channels is a one-step operation. Supply zero inputs to all transducers – this is typically done by placing the transducers away from all signal sources and by orienting the transducers to minimize the effect of gravity acting on any force or pressure transducers. Select `Record` and select `Null`. The recording duration will be increased to a large enough value to complete this operation. Select `OK` or press the keyboard space bar to start recording. The waveforms that appear in the scrolling display will be *raw* signal levels without any software offset correction, allowing you to adjust the amplifier panel offset controls for all channels to set the balance. This is the only time the waveform display will show uncorrected signal levels to allow setting the balance – the software offset correction from the last time you set the null is in effect under all other recording conditions. Select `Stop` or press the space bar when a stable, constant levels appear in all channels across the entire display window; this activates the dialog



At this point, the waveform displays show raw signal levels without any software null-offset correction. If you work the channel clicker to step through all of the channels, you will see that the offset correction applied to each channel reads zero. If the displayed waveforms show large offsets from zero, select `Cancel`, rerecord and select `Null` once again, this time adjusting the amplifier offet controls to obtain at least a coarse correction for the zero reading.

Select the `Set null offsets` button in the dialog to set the null-offset correction for all channels. The wave plots will all center at the zero level, and the offset reading for each channel will give the amount of correction applied in software. If the offset for any channel is more than 10 percent of the reported range, select `Cancel`, rerecord and select `Null` to reset the balance (amplifier offset control) for that channel. If the offsets are all within 10 percent of their respective ranges, select `OK` to accept the software null-offset corrections.

Once the null is set, and only if the null has been set, one can calibrate. Calibration is usually done on one channel at a time by supplying a known input level to that channel. The input level should be a least 10 percent of the peak value you want to measure to insure an accurate calibration. A force transducer may be calibrated by attaching a known weight; a pressure transducer may be calibrated by supplying pressure with a known height of a water column; a flow transducer may by calibrated by supplying a known airflow as measured by a ball-in-tapered-column flow meter.

Supply the known level to a transducer, invoke `Record`, select `Calibrate` and select `OK` to start recording. The waveform levels you see now have the software correction for null offset and read true zero when zero input is supplied to the transducers. Calibration requires supplying a known value different from zero in order to make an accurate determination of the transducer scale factor. You can make adjustments to the known level applied to the transducer while recording is taking place, but wait for the level to stabilize and show a constant trace across the entire waveform display window and select `Stop`. The calibrate dialog will appear – select the channel you are calibrating, enter the value of the known level, and select `OK`.

Once you have calibrated the system, you may also reset the null without having to redo the calibration because setting the null only changes the offset correction without changing the scaling of the data to read the calibrated range. Setting the null is an action that can be performed frequently as it can be performed by recording "open mike" with your transducers removed from your subject. Furthermore, nulls of many DC-coupled transducers can be extremely sensitive to temperature and other environmental effects caused by applying these transducers to your subjects while transducer sensitivies, the calibration range setting, are either less sensitive, less important, or less can be done about them if calibration takes a special setup. Furthermore, whether a signal is above or below the zero level may be important than the precise scaling of that deviation when measuring low level signals.

The rule is Null early and Null often.

3.5 Play menu

The `Play` menu has the commands

`Stop` – can also use the `Stop` button on the TF32 main window to stop playback.

`Interval` plays the waveform between the cursors of the selected channel.

`To end` keeps playing to the end of the waveform buffer; it scrolls the screen when it gets to the end of the displayed waveform interval.

`Sequence` plays the sequence of waveforms starting with the displayed waveform. The sequence is the set of waveforms in the same directory with the same extension taken in alphanumeric collating order; if a wave list was selected with `Files Open`, the sequence is the listed set of waveform files.

`Delay` allows setting pre and post playback intervals when playing a sequence of waveform files. The pre delay allows displays to be updated before playing the next waveform in the sequence on slow computers or when playing long waveform files. The post delay allows the display to dwell on the screen before selecting the next waveform in the sequence – useful for presentation purposes.

All waveform playback takes place through the sound card. Playback will be in stereo if you select the first of two waveform channels with the same attributes (file name, units, sampling rate). Otherwise playback will be monaural.

3.6 Filter menu

The filter commands are

`HPF` filters the displayed waveform with a zero-phase high-pass filter.

`Sine` adds a sine wave to the displayed waveform.

`Preset` applies the filter preset designated in the preset file to the waveform channels (requires Lab Automation level).

`Copy` copies a waveform channel into a new channel.

The `Preset` command is useful for applying the same signal conditioning filters used during automatic logging of waveform recordings (selected with `Files New session`). Invoke this command after retrieving the waveform with `Files Open`.

Automatic logging saves the raw acquisition waveform and filters the display waveform after saving and before logging measurements. When you open a logged waveform you have the option of applying the filter preset by invoking this command, you can apply different signal conditioning filters, or you can leave the waveform unfiltered if you so chose.

4 Placing waveform cursors

The placement of cursors on the waveform and time-frequency displays is controlled with the left mouse button. Clicking with the right mouse button on either of those displays brings up a menu for changing properties of that display.

Place and position the initial cursor by clicking the left mouse button over a waveform or time-frequency display and by dragging the cursor that appears. Place and position the final cursor by moving the mouse cursor to a new position and by clicking the left mouse button again. To select either of these cursors to make a measurement at that cursor position, position the mouse cursor over the cursor, and click the mouse cursor when you see the left-right arrow icon. Hold the left mouse button down to drag the cursor you have just selected. Remove a cursor from the display by dragging an initial cursor all the way to the left edge of the screen, a final cursor all the way to the right edge. If you want the final cursor to be placed first, select the button just below the menu with the right cursor-handle icon before placing any cursors.

Click on the up and down arrows that appear in the upper righthand corner of the TF32 main window to zoom out or zoom in to the waveform display. A horizontal scroll control appears after zooming in. Press the arrow buttons or drag the thumb button to scroll through the time-synchronized waveform and time-frequency spectrogram displays.

The waveform display settings – channel selection and scale of individual channels, cursor placement, and screen interval – may be controlled from the cursor keypad. Please refer to Table 1 where `Ctrl`-*key* means hold down the `Ctrl` key and press *key* once. The functions TF32 assigns to these keys are shown in Table 2.

4.1 Cursor keypad assignments

Channel selection and scale

`Home` Select lower-numbered channel (*e. g.* change from 2 to 1)

`End` Select higher-numbered channel (*e. g.* change from 1 to 2)

↑ Scale up display of selected channel

↓ Scale down display of selected channel.

Cursor placement

`Ctrl` ← Select initial waveform cursor for movement

`Ctrl` → Select final waveform cursor for movement

← Move the selected waveform cursor to the left

→ Move the selected waveform cursor to the right

Holding down `Ctrl` and pressing ← once (`Ctrl` ←) selects the initial waveform cursor. Pressing → moves this cursor to the right starting from the left edge of the screen. Each press of → advances the initial cursor an additional increment. Holding down `Ctrl` and pressing → once (`Ctrl` →) selects the final waveform cursor. Pressing → moves this cursor to the right starting from the position of the initial waveform cursor. Each press of → now advances the final cursor an additional increment.

Screen interval

`PgDn` Zoom in to the interval marked by waveform cursors

`PgUp` Zoom out to a wider screen interval

Home Ctrl-Home	↑	PgUp Ctrl-PgUp
← Ctrl ←		→ Ctrl →
End Ctrl-End	↓	PgDn Ctrl-PgDn
Ins		Del

Table 1: Keys of the keyboard cursor pad.

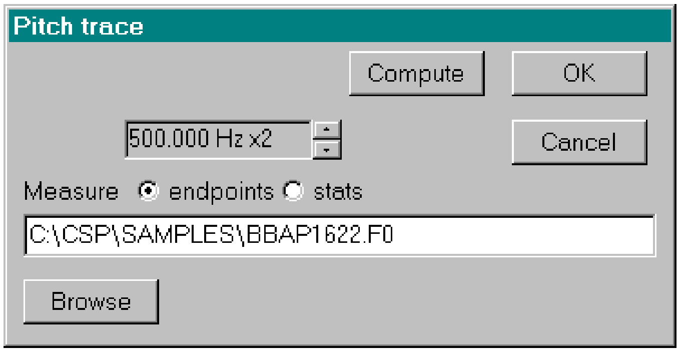
LowerChan	ScaleUp	ZoomOut
MoveCurLeft SelectInitCur		MoveCurRight SelectFinalCur
UpperChan	ScaleDn	ZoomIn

Table 2: TF32/CSpeech function assignments to keys of the keyboard cursor pad.

5 Displays

5.1 Pitch trace

The pitch trace display and its associated dialog compute, edit, save, and retrieve pitch. Selecting `Open View Pitch trace` activates the dialog



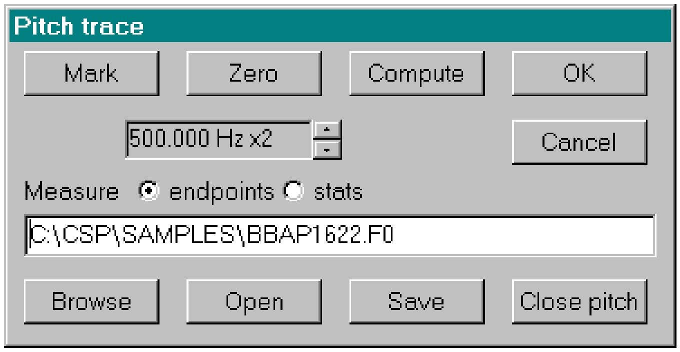
Selecting `OK` attempts to open a pitch trace from the file name shown in the edit box; if this file is not found, the pitch trace is computed. Selecting `Compute` computes a pitch trace, whether or not that file exists. Use the `Browse` button or directly enter into the edit box to change the file name selection.

Once a pitch trace is activated, it appears in an analysis trace window below the waveform windows. If open a new waveform file, the pitch trace window will try and read in the pitch trace from the same file name but with the `.F0` or other extension indicated in by the `Pitch trace` dialog. If the waveform files are in one directory but another directory has been specified for pitch trace files in the `Pitch trace` dialog, the pitch trace window will attempt to look for pitch trace files in that other directory. This situation arises when you have waveform files on a CD-ROM drive D: but save pitch traces to your drive C:. If the pitch trace window cannot find a pitch trace file, it computes the pitch trace of the new waveform.

A computed pitch trace appears in olive (yellow-green); a pitch trace retrieved from a file or a pitch trace that has been edited (you will be prompted to save it to a file) appears in cyan (blue-green). Filtering a waveform will automatically recompute a computed pitch trace; it has no effect on an edited pitch trace unless you select `Compute` from the pitch trace dialog to overwrite any editing changes and compute a new pitch trace from the filtered waveform.

While in CSpeech it was necessary to apply a high-pass filter before pitch analysis to eliminate low-frequency artifact, the high-pass filter in the TF32 pitch analyser is built-in. If you select `High pass` from the `Filter` menu, you are effectively high-pass filtering twice – please observe the effect on the pitch trace to determine if this helps your particular situation.

Selecting an active pitch trace from the `View` menu provides the dialog

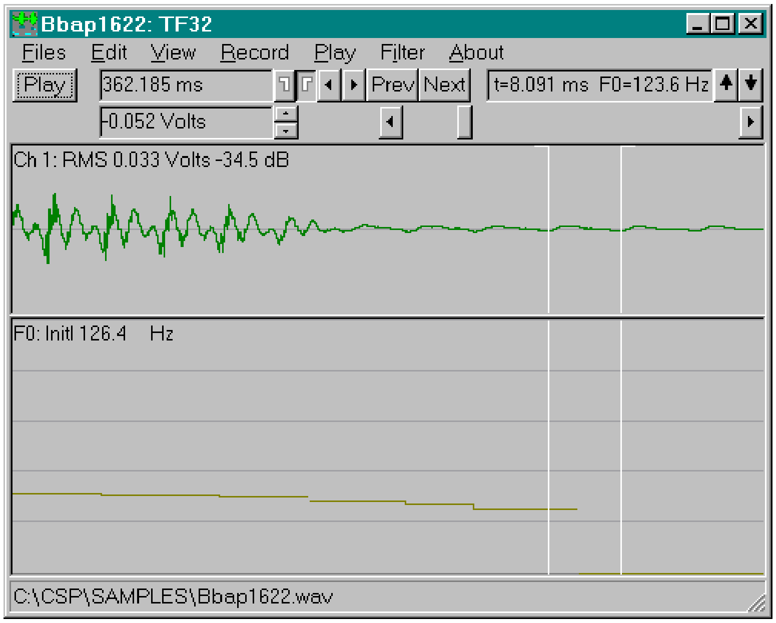


The range clicker allows changing the displayed pitch range. The `measure` buttons select numeric readouts that appear on the pitch trace display. The `endpoints` button selects pitch values at endpoints along with the change in pitch if pitch is displayed at both initial and final waveform cursor positions. The `stats` button selects mean, min, max, and standard deviation of pitch in Hz along with percent voiced, the fraction of the cursor-selected interval that had pitch values. `Open` tries to read in pitch values from a pitch trace, `Compute` overwrites any values with a computed pitch trace, while `Save` outputs pitch values to the ASCII (text) pitch trace file: the first column has times in ms when pitch changes and the second column has pitch values in Hz. The `Browse` button or edit control are used to change the pitch trace file name for retrieving or saving pitch. `Close pitch` closes the pitch trace display.

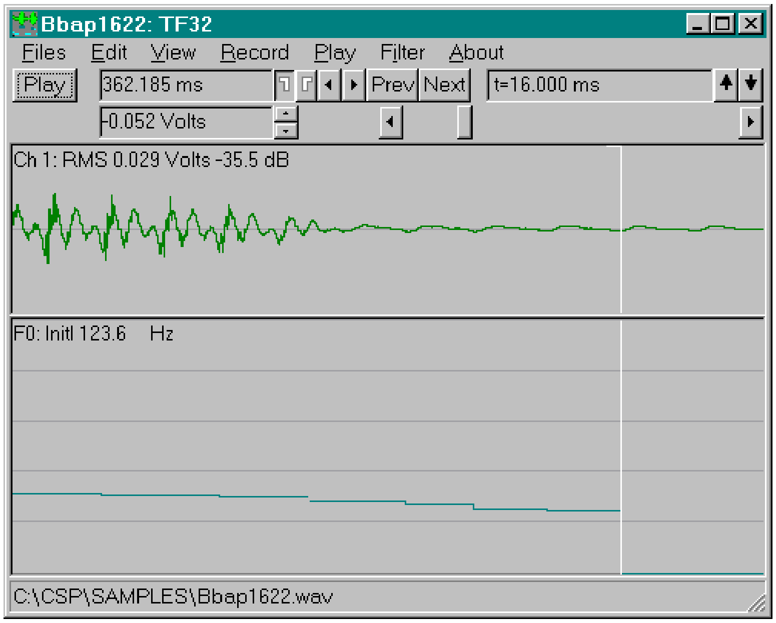
The algorithm used to compute pitch down samples by a factor of 2 for sampling rates greater than 14 kHz; it uses straight samples for lower sampling rates. It employs zero crossings and signal amplitude to make voice/unvoiced decisions on the original waveform, and it performs LPC inverse filtering on the downsampled waveform prior to cross-correlation analysis to reduce formant artifact on pitch tracking. It uses an initial 8 ms window that it correlates at lags ranging from .5 to 25 ms (pitch range is set to 40-1900 Hz), and on finding an initial pitch period, it uses a window that is adapted to the pitch period but limited to a minimum of 4 ms. The window is advanced with each pitch period, and the pitch trace changes on pitch period boundaries. The emphasis of this algorithm is the ability to track very rapid pitch changes associated with laryngeal stops, vocal fry, pitch doubling, and other voice breaks.

The editing control `Mark` computes the pitch frequency from a marked pitch period and adds it to the pitch trace; `Zero` zeroes out the cursor-selected interval of the pitch trace.

To mark a pitch period, zoom in to a small enough time scale to see individual pitch periods, and mark that period with the waveform cursors:



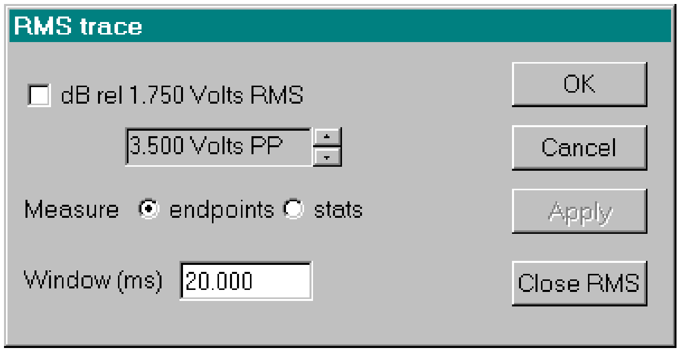
Next, select the pitch trace from the **View** menu and select the **Mark** button to produce



A pitch period-long interval has been marked with the corresponding pitch frequency, and the initial cursor has been advanced to allow marking the next pitch period. The pitch trace color changes from olive to cyan, indicating that you need to activate the pitch trace dialog and select **Save** to keep these editing changes.

5.2 RMS trace

The RMS trace display plots the RMS value of the selected waveform on either a linear or a dB scale. Selecting `View Open RMS trace` opens an RMS trace. Selecting that RMS trace from the `View` menu displays the dialog



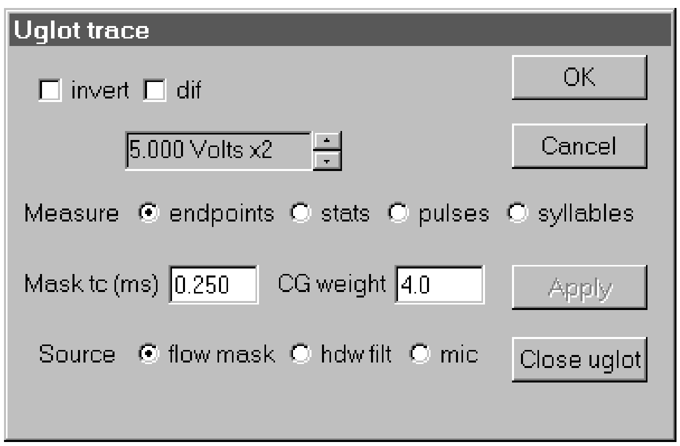
The dB checkbox allows selecting the dB scale. The range clicker selects the desired display scale of the RMS trace. The `measure` buttons select numeric readouts that appear on the RMS trace display. The `endpoints` button selects values at the waveform cursor positions. The `stats` button selects mean, min, max, and standard deviation of either RMS value or dB, depending on the dB checkbox. The `Close RMS` button removes the RMS trace display from the TF32 main window.

The RMS trace is automatically updated whenever you open or filter a waveform. If you select grid lines for the waveform with the `View Wave plot` dialog, those same grid lines appear on the RMS trace if is not in the dB mode. In the dB mode, the grid lines are at 10 dB spacing.

5.3 uglot – LPC inverse filter

The uglot display analyses a selected input waveform and plots the airflow waveform u at the glottis as estimated by an LPC inverse filtering algorithm, and it shows numeric measurements derived from the airflow waveform.

The uglot plot may be added to the TF32 display by selecting `View Open Uglot trace` from the main menu, and an existing uglot plot may be modified or closed by selecting that trace from the `View` menu. The uglot analysis can be applied either to the raw flow signal measured from a circumferentially-vented flow mask, from the glottal flow signal from a hardware inverse filter (no LPC filter or mask correction), or to a low phase distortion microphone signal – the type of source as well as the waveform channel containing the source signal are selectable from the dialog that appears when adding or modifing a uglot trace.



The `invert` check box inverts the input waveform prior to analysis – selecting `invert` not only affects the orientation of the uglot trace, it also affects the numeric analysis obtained from that trace, so check or uncheck `invert` so low flow is down, high flow is up. The `dif` check box takes the first difference $u[n] - u[n-1]$ between successive flow samples to give the flow derivative or voice source signal.

The range clicker changes the magnification of the flow waveform on the display, but it does not change the underlying data values. Units other than volts (such as ml/s) can be specified in an ASCII preset file as part of the installation of TF32 Lab Automation, and depending on how TF32 is configured with a preset file, calibration may be enabled in the Record dialog for setting the units range and offset against a known flow standard – see page 20 for more information about recording a calibration setting and page 72 for more information of calibrating flow levels. The channel clicker only appears if there is more than one waveform channel to chose from and allows selecting the waveform to be analysed.

The `Measure` buttons allow selecting the numeric readout that appears on the uglot plot. Like all numeric readouts selected for waveforms or for waveform analysis traces, these numbers may be pasted to the Windows clipboard with `Edit Copy readouts to clipboard`, or you can save readouts in batch mode. For batch mode analysis, add entries to a wave list by opening a waveform file, locating the waveform cursors on the desired interval and invoking Edit Add index to wave list for each entry. When you have made the required entries, open that list file with `Files Open`, verify that the uglot trace has the desired `Measure` and `Source` selections, and then invoke `Edit Measure sequence`.

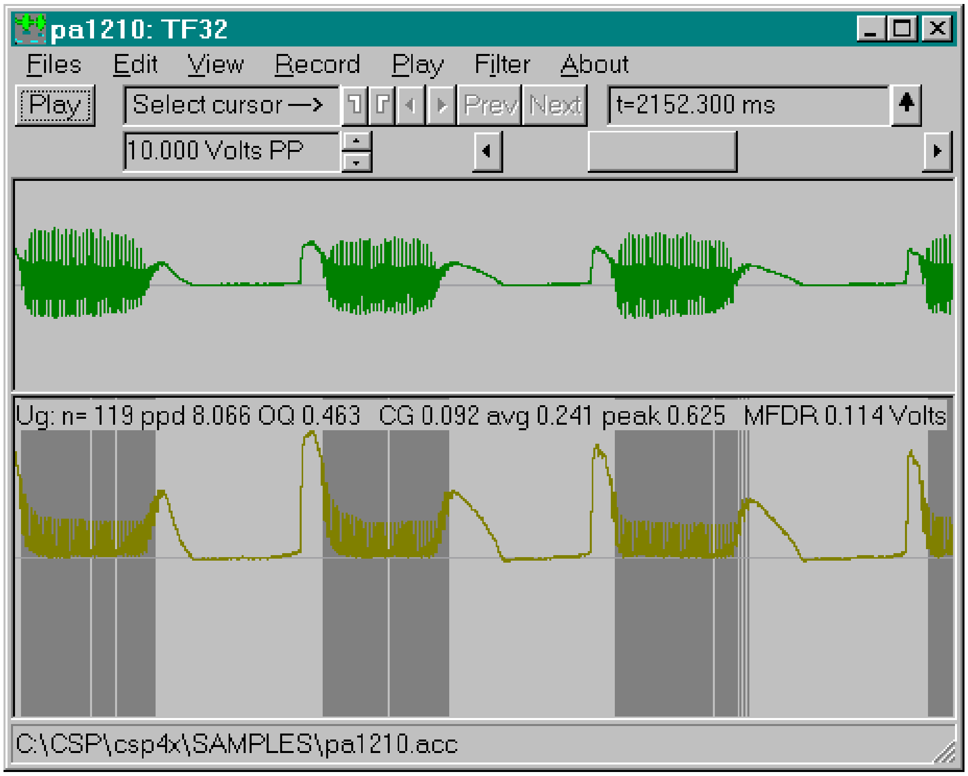
Among the `Measure` selections, `endpoints` displays uglot values at the initial and final waveform cursor positions, and `stats` displays the mean, min, max, and standard deviation of uglot samples for the cursor-selected interval. The selections `pulses` and `syllables` report averages on measures that apply to individual glottal waveform cycles. The selection `pulses` marks the closed glottis interval and reports averages of all glottal cycles it identifies in the cursor-selected interval; the selection `syllables` only marks the closed glottis interval for the central sections of complete syllables it identifies in the cursor-selected interval. A syllable is identified if the gap between glottal pulses preceding and following is greater than 50 ms. A syllable is measured if it is at least 200 ms long, and the 50 ms voice onset and voice offset durations are excluded to analyse a central portion that is at least 100 ms long.

The `Mask tc` edit box pertains to the `flow mask` selection of inverse filter source. It specifies the time constant τ of a circumferentially vented mask as reported by Equation 9 of M. Rothenberg (1977), Measurement of airflow in speech, *JSHR* 20 155-176. The default setting of .25 ms corresponds to a 10 cm sq mask, a product that can be purchased from Glottal Enterprises (Syracuse, NY).

The LPC analysis used to inverse filter each glottal cycle starting with the onset of glottal opening and ending prior to the next opening onset is updated with each pitch period. The LPC algorithm is a weighted least squares that gives a unit least squares weight to the entire pitch period interval and a least square weight of 16 (a waveform magnitude weight of 4) to a half pitch period interval located after the time of maximum voice source excitation. The `CG weight` edit box allows changing the magnitude weight from the default value of 4 to 0 (least-squares minimization over whole pitch-period interval only) to a large value like 100 (effectively least-squares over closed-glottis interval only). This same algorithm with the default magnitude weight of 4 is used in formant determination; it is used here because it reduces formant ripple compared with pitch-period least squares, and it produces somewhat fewer spurious analysis frames on nasals and voiced obstruents as LPC analysis restricted to the closed glottis interval.

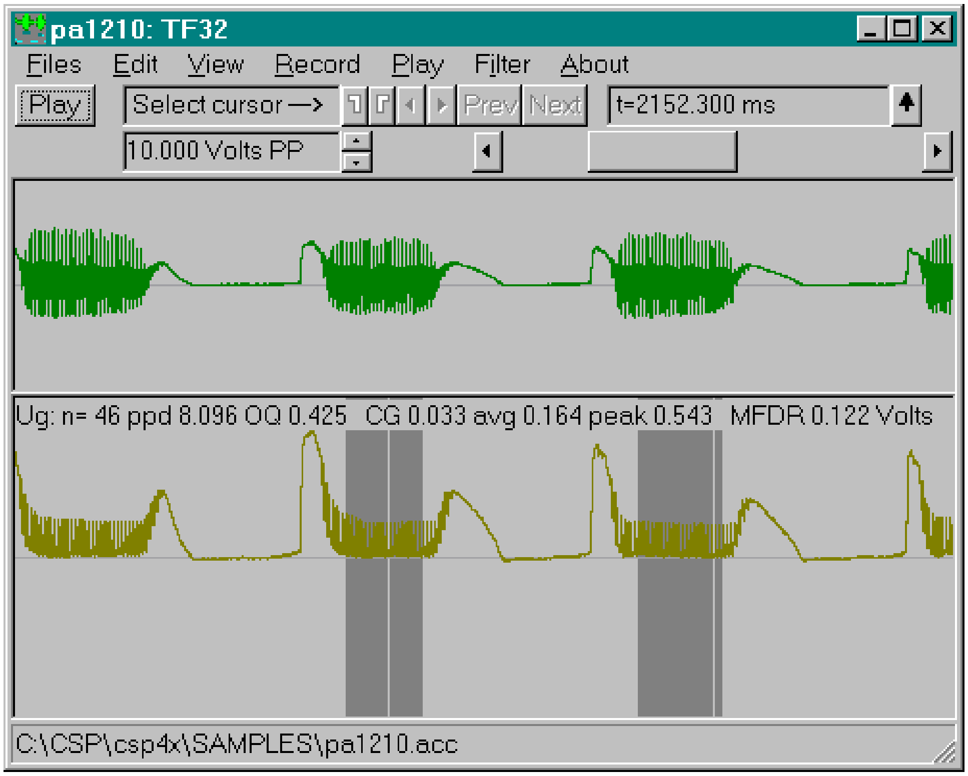
The following examples are from a flow waveform recorded with a Glottal Enterprises circumferentially-vented mask.

This figure shows the `pulses` selection. The dark bands are regions where glottal pulse have been identified. The cursor-selected interval is by default the entire screen width as no waveform cursors have been placed.



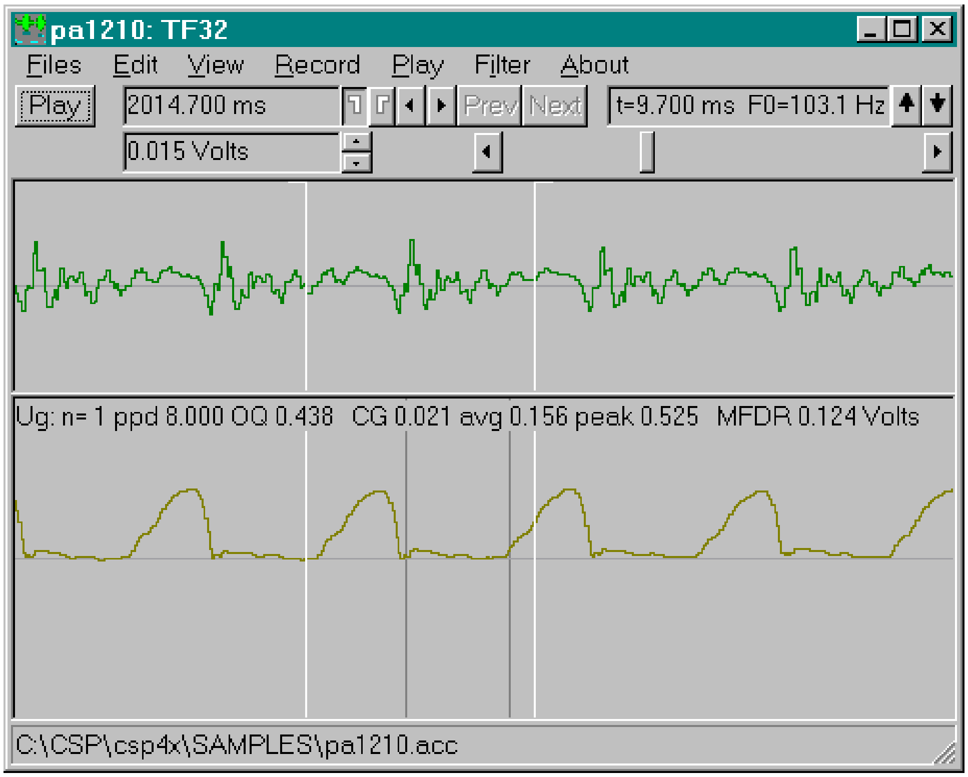
The numeric readout reports `n`, the number of identified glottal pulses, and averages for all the identified pulses for `ppd`, glottal cycle duration in ms for the identified pulses, `OQ`, the open quotient or fraction of cycle time that the glottis is presumed open, `CG`, the average flow in the identified closed-glottis interval, `avg`, the average flow over an entire cycle, `peak`, peak flow for a cycle, and `MFDR`, the maximum flow declination rate – the maximum value of the difference between flow samples $u[n-1] - u[n]$ that occurs on the downsloping side of the glottal pulse while the closure that initiates vocal tract formant oscillations is taking place.

The next figure shows the `syllables` selection. Only the central portions of the middle two syllables have been selected. The last syllable is not complete, and the selected interval does not include the required 50 ms minimum inter-syllable separation to accept the first syllable.



Note that fewer glottal cycles have been reported, and reported `CG`, `avg`, and `peak` flows have diminished on account of excluding the voice onset and offset intervals.

Zooming the time scale of the display, selecting the displayed `uglot trace` dialog from the `View` menu to magnify the uglot display and selecting the `pulses` measurement mode, and positioning the waveform cursors around one complete glottal pulse cycle results in

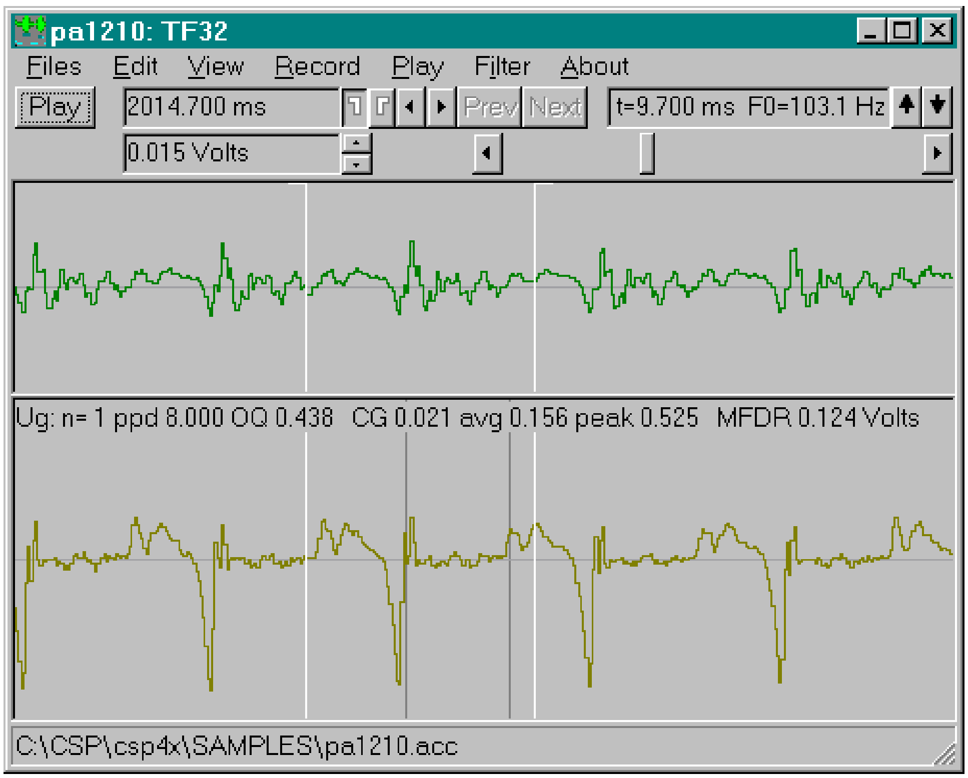


It is important to select `pulses` in this example because `syllables` will not mark an individual pitch period cycle. The glottal pulse cycle as reported in the numeric measurements starts at the initiation of glottal opening and ends at the end of glottal closure. The estimated closed glottis interval is marked with a pair of gray cursors. If you zoom out the display, you will see those marker cursors merging into gray bands as seen in the previous two figures.

The closed-glottis intervals cursors indicate the glottal pulses that have been identified and show what has been selected as the nominal closed portion of the glottal flow waveform cycle for the numeric measures. While you cannot override or edit these selections, you can verify the operation of the automatic selection and reject analysis results not meeting established criteria.

The closed glottis onset is specified by looking at the flow declination signal (first difference u[n-1] - u[n]) and finding the declination peak that corresponds to the maximum downslope of the glottal pulse during glottal closure. The closed glottis onset is the first point in time where flow declination crosses below 20 percent of the peak declination. Flow declination is used to determine closed glottis onset on account of the strong declination during glottal closure. The closed glottis offset is obtained by taking the peak *flow* (not the flow declination or derivative) and by finding the first point preceding the following glottal flow peak that is below the 12 percent of that peak. The actual flow is used to determine offset because the flow declination is weak during glottal opening.

These rules for finding a first level crossing either following or preceding a peak are meant to avoid ambiguity when the closed glottis interval is not perfectly flat and may have multiple level crossings unless the levels are set very high in relation to the peaks. The effect of these rules may be observed by comparing the preceding figure with the following figure, where the `Dif` checkbox has been selected in the uglot dialog and the uglot trace has been further magnified. The `Dif` checkbox takes the first difference $u[n]-u[n-1]$, negative of the flow declination, and the flow declination peak is the sharp downward spike



Note that the numeric readouts have not changed because checking the `Dif` box only changes the displayed uglot trace; the numeric readouts for the `pulse` and `syllables` modes report the same glottal flow and glottal flow derivative (MFDR) numbers as with `Dif` unchecked.

Flow sources

The uglot analysis can work with any of three source waveforms selectable from the uglot trace dialog.

The `flow mask` selection is for a circumferentially-vented mask with fine wire mesh covering the venting ports. This type of mask has a time constant τ associated with as reported by Equation 9 of M. Rothenberg (1977), Measurement of airflow in speech, *JSHR* 20 155-176. The time constant is related to the acoustic inductance of the mask ports, producing a corner frequency, below which the mask measures flow and above which it measures pressure. The Rothenberg (1977) paper reported on a 28 cm sq mask, giving a time constant of .42 ms. The `flow mask` setting of the uglot analyser in TF32 applies a time constant correction of .25 ms, which corresponds to a 10 cm sq mask, a product that can be purchased from Glottal Enterprises (Syracuse, NY).

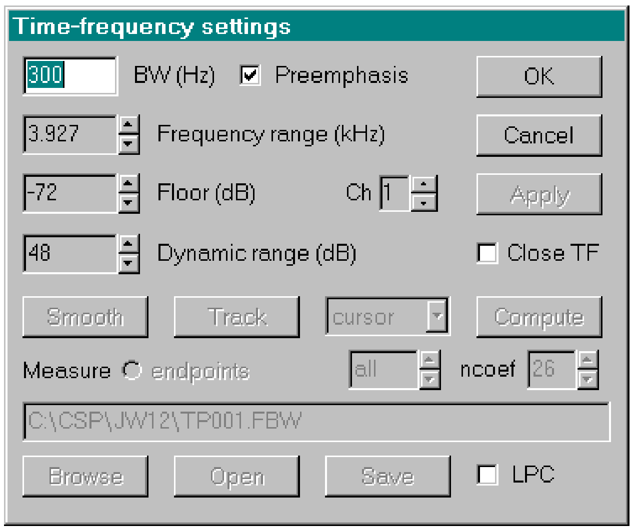
The `hdw filt` selection is applicable to the output of a hardware inverse filter device. It is assumed that both the mask correction characterized by τ and the inverse filter have already been applied so neither of these corrections are applied in software. This selection provides a direct pass through between the recorded waveform and the uglot display, although the uglot display still segments pitch periods, identifies glottal pulses, and reports numeric results.

The `mic` selection is applicable to a pressure signal recorded with a microphone. Meaningful open quotient numbers require selecting a microphone and preamplifier circuit with a high-pass cutoff frequency below 2 Hz. Such a signal will have substantial low-frequency artifact from air conditioning and other sources of air movement; the uglot analysis automatically applies a zero-phase high pass filter to the signal to remove this artifact without introducing phase error. Since a microphone signal blocks an DC component, it is impossible to determine absolute levels of flow as in the determination of leakage flow in the closed-glottis interval. The numeric readouts `pulses` and `syllables` only report the flow differences `peak-avg` and `peak-CG` when the `mic` source is selected.

5.4 Time-frequency spectrogram

TF32 allows up to two time-frequency gray-scale spectrogram displays with selectable LPC formant overlay. Select a new spectrogram by invoking `View Open TimeFreq`; select one of two active spectrogram displays by invoking `View TimeFreqA` or `View TimeFreqB`.

Selecting a spectrogram activates the dialog



The top half of the dialog controls the spectrogram; the bottom half controls the selectable LPC formant track overlay. The channel clicker (visible only when more than one waveform channel to chose from) selects the waveform channel analysed by both the spectrogram and the formant overlay. The `Close TF` checkbox grays all the other controls when selected – click `OK` to close the spectrogram display. The `LPC` checkbox activates the LPC formant track overlay and activates the controls on the bottom half of the dialog.

5.4.1 Spectrogram settings

The `BW` edit box is for entering the spectrogram analysis bandwidth in Hz. Bandwidth values have widespread meaning among speech scientists and clinicians. A value of 300 Hz is a commonly-used wide-band setting in speech analysis, used to show formants as broad bands, while 45 Hz gives a narrow-band spectrogram, showing individual harmonics of the fundamental. For high-pitched talkers, a bandwidth of 450 Hz may be required as a wide-band setting on account of the wider spacing of the harmonics one is trying to average out to show the formants. The spectrogram calculation uses a Hamming window: with the 300 Hz setting; the window spans 6.7 ms; at 45 Hz, window spans 22 ms.

After entering a number in the edit box, press `Apply` to see changes to the displayed spectrogram without exiting this dialog; press `OK` to make the changes and exit the dialog.

The `Frequency range` clicker determines the upper frequency limit of the displayed spectrogram. When selecting an upper frequency limit for the display greater than the maximum spectrum extent of half the sampling frequency, the out-of-range part of the display will show a dark gray band.

The spectrogram is made up of a sequence of spectra shifted in time, where the vertical axis of the display is the frequency axis of each spectrum. The available frequency ranges are chosen so that vertical screen pixels fall exactly on spectrum samples at 1, 2, 4, or 8 sample spacing, depending on the frequency range selected. The display shows actual spectrum values without intervening pixel interpolation. This also means that the available frequency ranges occur in big steps, and the available frequency ranges depend on the video monitor resolution and size of the TF32 window on the display. Hence TF32 gives greater priority to spectrogram accuracy, resulting in frequency ranges that may not be round numbers.

The `Floor` clicker sets the minimum dB level – lowest level signal – visible on the spectrogram display. Operating this clicker shows changes to the spectrogram display underneath the dialog window, useful in getting the desired setting.

The `Dynamic range` clicker sets the dB span of the display. A low value of 32 dB gives a very high contrast display that may reproduce better in print. A high value of 64 dB actually may show more information because of the wider gradation of signal levels, but it produces a low contrast display that may not reproduce well. The default value of 48 dB is a workable compromise between extremes.

The `Preemphasis` checkbox selects a 6 dB per octave increase in the high frequencies, obtained by computing the first difference of the waveform prior to computing the spectrum. The default is preemphasis on, which is commonly used for speech analysis. Turning preemphasis off computes the spectrum from the original waveform without the differencing operation.

5.4.2 LPC formant track settings

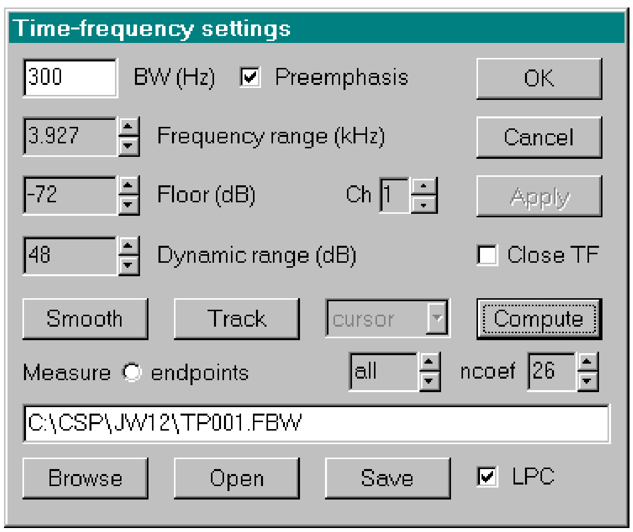
Select the `LPC` checkbox to enable the LPC formant track overlay. The software first tries to retrieve formants from a `.fbw` file. A grayed `ncoef` clicker indicates that formants were found in a file – the number of LPC coefficients is fixed and can only be changed by recomputing formants. An active `ncoef` clicker indicates that formants have been computed – the formants are recomputed for each press of that clicker. To open formants from a different file than the one indicated in the edit box, edit the file name, or search for a formant file by selecting `Browse`, and select `Open`. Select `Compute` to reenable the `ncoef` clicker, allowing formants to be recomputed.

The LPC formant analysis determines the location of each voiced speech pitch period, computes LPC coefficients, solves for the frequencies and bandwidths of the LPC poles, and makes an initial determination of which poles are formants by setting a bandwidth threshold.

The LPC algorithm is a weighted least squares that gives a unit least squares weight to the entire pitch period interval and a least square weight of 16 (a waveform magnitude weight of 4) to a half pitch period interval located after the time of maximum voice source excitation. This method offers some of the formant accuracy advantage of least squares analysis restricted to the closed-glottis interval while preventing the LPC spectrum for departing too greatly from the stationary speech spectrum as computed over an entire pitch period interval.

The resulting LPC pole tracks are plotted in color overlying the spectrogram. Poles identified as formants are red, non-formant poles narrower than 500 Hz bandwidth are yellow, and poles between 500 and 1000 Hz are green. Selecting `endpoints` on the spectrogram dialog enables a readout of formant values overlaying the spectrogram display showing formant values at a pair of cursor positions.

The time-frequency spectrogram dialog with LPC formant track overlay enabled looks like



The `Smooth` and `Track` buttons along with the list boxes showing `cursor` and `all` allow editing the formant tracks while the `Save` button saves editing changes to a file.

The `Smooth` and `Track` buttons apply an automatic correction to the LPC poles and formant tracks for a cursor selected interval. You may need to exit the time-frequency dialog, drag the mouse across either the waveform or spectrogram display to place time cursors, and reselect the time-frequency dialog from the `View` menu.

The `Track` button refines the designation of LPC poles as formants by applying a dynamic programming algorithm that takes into account likely formant frequency and bandwidth values as well as temporal continuity of both frequency and bandwidth values of poles belonging to a formant track. It does not change any of the LPC poles – it only refines the assignment of poles to formants. Another feature of the algorithm is that it allows formants to start and end – it does not force an assignment for formants F1, F2, and F3 for every analysed pitch period. Depending on voice source and nasalization characteristics, formants can disappear and reappear from the time sequence of LPC poles, and forcing the selection of a formant at every frame can lead to errors.

The `Smooth` button recalculates LPC poles by applying time smoothing and then reassigns formants to poles using the same procedure as `Track`. Poles are calculated by averaging log-area ratio LPC parameters over pitch periods within an interval that 1) can range up to 50 ms, 2) is centered on the pitch period for which formants are reported, and 3) is constrained not to overlap outside the cursor selected interval. Log-area ratio coefficients can safely interpolate LPC coefficients without resulting in unstable LPC poles.

You may want to apply this formant synthesis to cursor-selected intervals so that you smooth vowel regions but do not smooth across vowel-nasal boundaries. It is worth experimenting with the selection of intervals to get results consistent with the underlying speech spectrogram.

The two list boxes showing `cursor` and `all` are the controls for hand correction of formant tracks. Voice source and nasal zeroes can cause LPC formants to "drop out" and hand editing may be indicated, depending on your criterion for measuring formants. The `all` list box allows selecting the formant (F1, F2, or F3) to edit – only that formant will appear in red. The `cursor` list box specifies the editing mode. Select a formant first to enable the editing mode list box. Select `Exit` to close the dialog, use the mouse to edit the formant track, and reselect the dialog from the `View` menu in order to make further selections or select `Save` to save the changes.

Editing is done by placing the mouse cursor over the spectrogram display, pressing the left mouse button, and "painting" with the mouse to effect the desired formant change. The active editing modes are `select`, `trace`, and `erase`, and in those modes, a cross hair appears as the mouse cursor when that cursor is positioned over the spectrogram. The mode `cursor` disables editing and allows use of the mouse to position interval selection cursors over the spectrogram display. If you are in an editing mode, you can still position time cursors by making selections on the waveform display, but remember that clicking on the spectrogram display with the mouse will change formant values. You may also want to zoom the display time interval or change the spectrogram frequency range to have finer control over positioning formants with the mouse.

The `select` mode constrains a formant to one of the computed LPC poles. Use this mode if the formant error is one of mislabelling. By dragging the mouse, you can "paint" the desired LPC pole track to select it as the desired formant. If you paint a pole as F1 that had been previously assigned to F2, F2 is no longer assigned, so you will have to remember to select F2, assign it to a pole track, and then do the same thing for F3.

The `trace` mode allows positioning a formant "free hand" without regard to the

LPC pole frequencies. Use this mode in the case of formant drop outs. Also consider the trying the `Smooth` button on cursor selected intervals before reverting to hand tracing. The trace mode also allows extending formant tracks into unvoiced regions where the automatic method does not compute LPC poles or into voiced regions were the pitch period locator has drop outs. The `erase` mode allows removing a formant track, either to correct errors of the automatic method or to correct editing mistakes.

5.4.3 Contents of the `.fbw` formant track file

The default dialog setting saves and retrieves formant tracks from a numeric text file with the same name as the speech waveform but with the `.fbw` extension. The `.fbw` format is an elaboration of the format used with CSpeech and is both upward and downward compatible: TF32 .fbw files display in CSpeech, CSpeech .fbw files display in TF32.

The extension name originally stood for (LPC formant pole) frequencies and bandwidths, but bandwidth values somehow never made it into the file, but the name stays for compatibility reasons.

The CSpeech `.fbw` format has columns in the order time value, F1, F2, F3, and dB value. Time is in ms, frequency in kHz. The TF32 `.fbw` format adds columns for pitch period duration in ms and number of averaging frames. The purpose of pitch period duration and number of averaging frames is to allow TF32 to read back formants from the `.fbw` file and recompute the LPC poles that had previously been used to determine those formants, which allows comparing formants to LPC formants for further editing.

The number of averaging frames keeps track of the use of the `Smooth` function, even if `Smooth` has been applied to multiple intervals. This number is the number of preceding and following pitch period frames averaged to determine LPC coefficients for a given pitch period frame. There are always the same number of preceding and following frames, so this information is coded as one number. A value of 0 means only the current frame was used to compute LPC; no preceding or following frames were averaged. Since the initial frame in a file has a number of 0 (there can be no preceding frames hence there are no following frames), the number that appears in the initial frame of the file is set to the number of LPC coefficients – it is a good place to save `ncoef` without adding another line or another column.

So the augmented `.fbw` file is really a data compressed format: frame duration and averaging number plus computation substitute for storing a lot of numbers in the file. The essence of data compression is to substitute computation for storage; the speed of modern microprocessors make this a useful tradeoff.

One limitation of this system is evident if you compute formants after applying the high-pass filter to a waveform (select `Filter HiPass` from the menu), but retrieve formants after opening the same waveform without reapplying the high-pass filter. You may see the yellow pole tracks poke out from under the red formant tracks in places, indicating that the recomputed LPC coefficients have shifted. Be sure to reapply any waveform filters when reediting formant tracks to insure that the LPC coefficients are the same.

It is safe to compute formants from a waveform that has not been high-pass filtered because the LPC computation applies its own high-pass pre filter to remove any DC offsets and low-frequency artifact that may affect the computation. Applying a high-pass filter to that waveform means the LPC computation sees a signal that has been high-pass filtered twice, which is OK if you need to high-pass filter a waveform for dB readings or other analyses and if you reapply the same high-pass filter when reediting formant tracks.

A major change between TF32 and CSpeech is that TF32 employs an algorithm to locate pitch period epochs and computes formants pitch synchronously while CSpeech relied on fixed duration analysis frames. The CSpeech `.fbw` file has formant values at equal time intervals. The TF32 `.fbw` file segments time into 8 ms intervals during unvoiced segments to allow for editing formant tracks into those segments; it segments time into pitch period intervals during voiced segments. Please take this into account when post processing TF32 `.fbw files`.

5.4.4 Measuring formant endpoints

While the `.fbw` file may be read in by spreadsheets and software that accepts numeric text from files, often times the data of interest is a particular formant transitions, and TF32 provides a way of measuring and saving formant values at the two endpoints of such transitions.

Selecting `LPC` followed by selecting `endpoints` from the time-frequency dialog enables the numeric readout of formant frequencies F1, F2, and F3 at initial and final waveform cursor locations along with initial-final formant frequency differences.

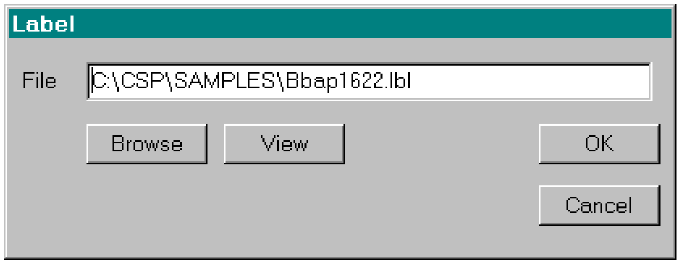
One option is to invoke `Edit Copy readouts to clipboard` for each reading and then paste those numbers into a text editor or spreadsheet window you have open the same time you are running TF32. Windows allows multiple applications to be active and allows pasting data between applications.

Another option is to mark time intervals of one more more files for which you have computed formants and saved any editing changes, and to add each time interval to a wave list file (`.lst`) with the `Edit Add index to wave list` command. The `Edit Add index to wave list` dialog has a `View` button that allows you to inspect the list of time indices and file names to verify that you have saved the ones you want.

Next, open that `.lst` file with the `Files Open` command to bring up the first interval of the first file in that list. Select the desired readouts – be sure formant readouts are activated for the spectrogram display and any other readouts you want for pitch or other analyses are activated – and apply any desired filters. Invoke `Edit Measure sequence` to automatically step through each list entry, apply any filters, and save numeric readouts for each list entry to a text file.

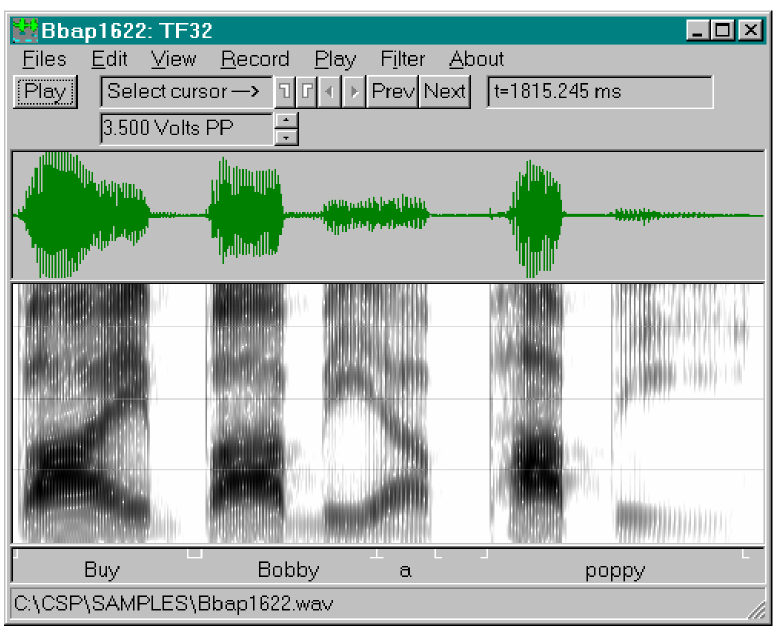
5.5 Label display

The command `View Open Label` that places the label display on the TF32 main window activates the dialog



The `Browse` button or the edit box can be used to change the default file selection or to keep that file selection while retrieving and saving labels from a different directory. The label file can go into a different directory from the waveform file to help with the situation where the waveform file is on CD-ROM drive D: and labels are saved to drive C:. The `View` button activates an editor window that displays the contents of the label file, if the file exists, and allows hand correction of entries. Each line in the label file is an initial time, a final time (units are ms) followed by a text label.

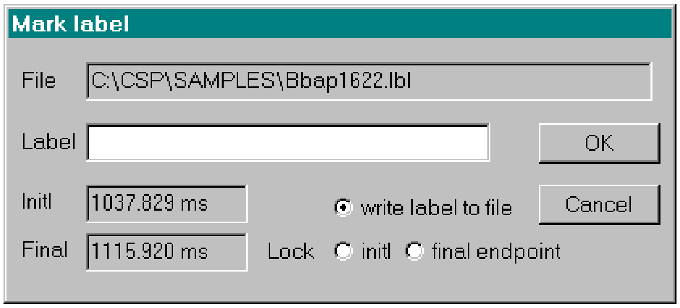
Selecting `OK` activates the label display on the TF32 main window:



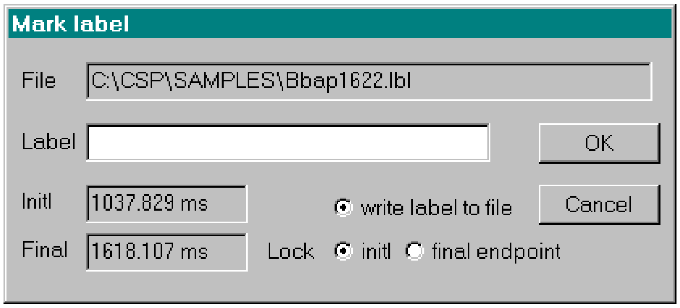
If the label file does not exist or is empty, the label display will be empty. The

`View` menu will now have a `Label` command for viewing the contents of the label file, changing the label file name or directory, or to close the label display; the `View` menu will also list a `Mark` command for adding labels which appear in the label display and are immediately saved to the label file.

The mark a label, invoke `View Mark` to activate the dialog



If the waveform cursors are at precise locations you want to mark with a label, enter that text label in the edit box and select `OK`. That text will appear in the label display along with handle cursor tic marks indicating the labelled interval, and the label entry will be added to the file. Sometimes, you may need to zoom in to two separate locations for precise placement of the initial and final cursors. If the initial cursor is in the right place, select the `Lock initl` button and select `OK` to exit the dialog. Zoom to the other location, place the final waveform cursor, and invoke `View Mark label` to get the dialog back.



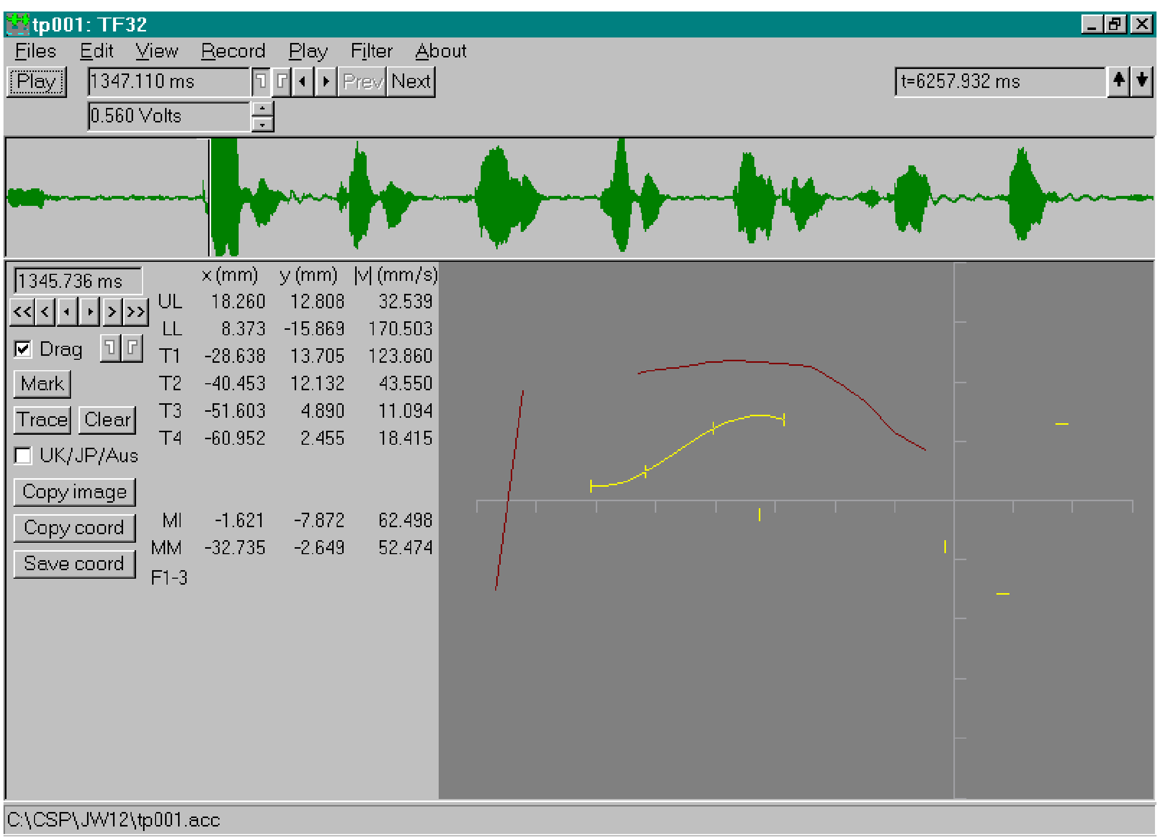
You will see that both the `write label to file` and `Lock initl` selections are displayed. This means that the initial cursor had been "locked down" to the position it had the last time the dialog was active when you had selected `Lock initl`. The final cursor position is whatever appears in the waveform display. At this point, enter the text label and select `OK`. If the label locations indicated by the time tags in the dialog are not correct, uncheck `Lock initl` (this grays out the text edit box) and select `OK` to start over.

5.6 x-y articulatory display

The x-y articulatory display requires the Demo level of TF32, which may be downloaded from `http://www.medsch.wisc.edu/~milenkvc/tools.html`. Information on obtaining the CD-ROMS of the University of Wisconsin X-ray Speech Production Database is found at `http://www.medsch.wisc.edu/ubeam/`.

The x-y display plots the positions of articulatory markers in the midsagittal plane as measured by a computer-controlled x-ray system. When TF32 is viewing an acoustic waveform from a file called `TP001.acc`, the x-y display plots the articulatory data found in a file called `TP001.xyd`. The web page `http://www.medsch.wisc.edu/~milenkvc/tools.html` contains detailed information about the format of the `.xyd` file. The x-y display can be used with measurements from a magnetic articulograph as well, if those measurements can be stored in that format.

Opening waveform record `TP001.acc` from subject directory `JW12` from the UW X-ray Speech Production Database CD-ROM Volume 1, selecting the x-y display by invoking `View Open XY` from the main menu, clicking the `Drag` checkbox on the x-y display control panel, dragging with the left mouse button over the waveform plot to 1347 ms, and maximizing the TF32 window to full screen gives the following display



The left-hand part of the x-y display control panel has buttons for operating on the display. The right-hand part of the control panel gives a readout of pellet marker positions and velocity magnitudes at the selected time location. The x-y plot is to the right of the control panel.

Time readout

The time readout in the upper left of the x-y display gives the time position of the articulatory display, and it moves in steps of 6.866 ms, the interval between articulatory samples. That time position is also indicated by the black *play* cursor on the waveform plot. Note that the waveform cursor time readout just below the TF32 menu shows 1347.110 ms while the x-y display shows 1345.736 ms and that the black play cursor is not aligned with the waveform cursor (white cursor with left-facing handle placed by dragging with the left mouse button). The black play cursor on the waveform plot shows the precise location of the x-y display articulatory sample, which can be slightly offset from the white cursor selecting a waveform sample on account of the 6.866 ms interval between samples of the articulatory markers.

Coordinate readout

You must maximize the TF32 window to make the coordinate readout visible. Also note that selecting the x-y display hides the time-frequency display unless the formant overlay is active; select `View Open TimeFreq` from the menu if you want to bring the time-frequency spectrogram back. If you reactivate the time-frequency display and select the formant track overlay, the formant frequencies for the selected time location will appear at the bottom of the list of articulatory coordinates.

You may observe instances where one or more coordinates are missing – you will see that the corresponding pellet is also missing from the x-y display. Instances where the computer-controlled x-ray system is not able to locate one or more pellets are not displayed; these instances are marked in the pellet position data with a large number denoting a missing or bad data value. The x-ray system can fail to located a pellet for a number of reasons: the pellet is blocked by an x-ray opaque dental filling, a cosmic ray photon saturated the x-ray detector.

The x-ray system can mistakenly superimpose two pellets on the same x-y position; this can occur when the system loses track of one pellet and mistakenly requires track on top of another pellet. The coordinate readout reports superimposed pellets in red. If you are observing redacted data such as the UW X-ray Production Database CD-ROMs, you will not see this condition. You may see this condition with unredacted data collected with the x-ray system.

Articulatory plot

The plot to the right of the control panel shows the back pharyngeal wall in red to the left, the palate in red towards the top, and four pellet markers on the tongue joined by a smooth curve in yellow towards the middle. The four tongue points are joined by an arc of linearly-varying curvature according to the Curvic algorithm (Milenkovic, V., and Milenkovic, P. H. (1996). Tongue model for characterizing vocal tract kinematics, in Advances in Robot Kinematics, J. Lenarcic and V. Parenti-Castelli eds., Kluwer, 217-224). The tongue in this example is positioned at the /r/ in the word "problem" and shows the serpentine shape characteristic of /r/ for subject JW12 in the Database. Markers for the lower jaw are seen below the x-axis grid line; markers for the lips are seen to the left of the y-axis grid line. The grid intersection is the (0, 0) point: x-values to the left and y-values to the bottom both read negative, values to the upper right both read positive. The articulatory manikin is facing right; select the `UK/JP/Aus` checkbox (countries that drive on the opposite side of the road) to change the display to facing left.

The colors of red for palate trace, yellow for tongue outline are valid when Windows is in the 256 color mode. If other colors appear, you may want to switch the Windows graphics display to the 256 color mode: select the `Display` icon from the Windows Control Panel to make this change.

Control panel buttons

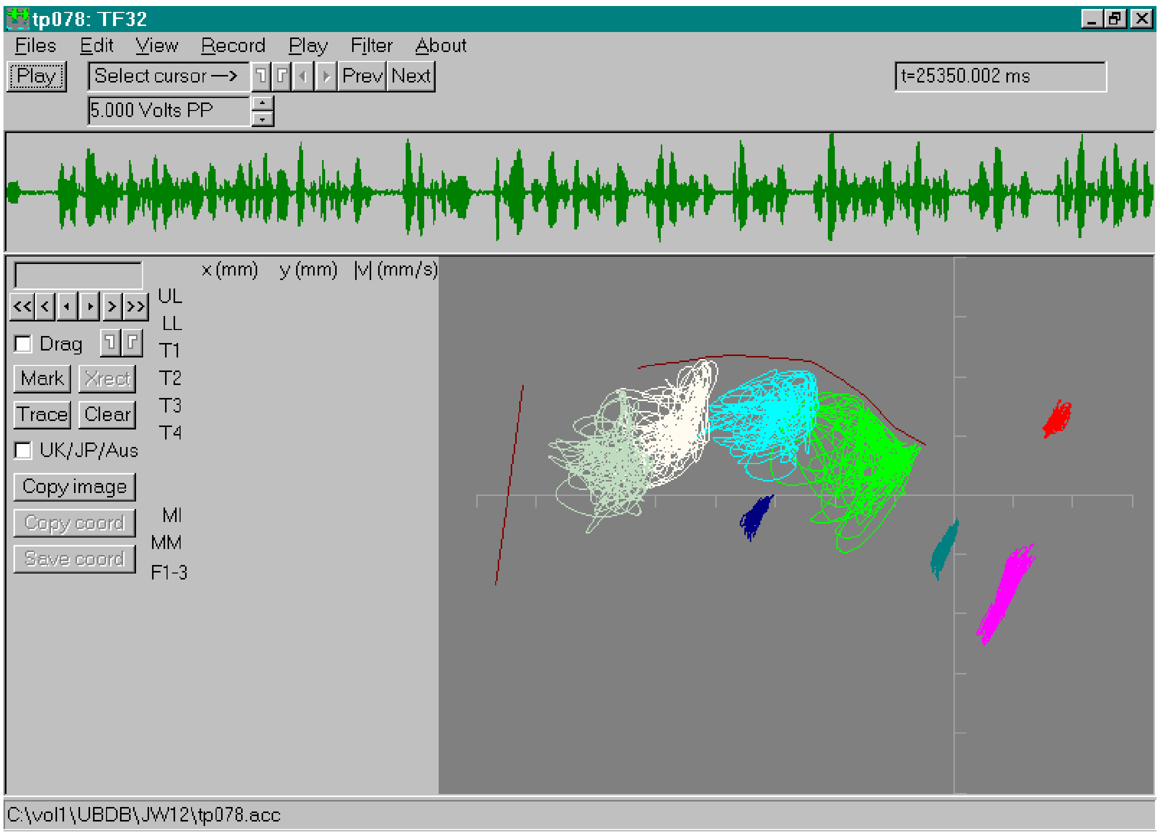
The arrow buttons, `Drag` checkbox, and white handle cursor buttons below the x-y display time readout set the time position of the articulatory display. The outward pair of arrow buttons, `<<` and `>>`, set the articulatory display at the initial and final positions selected with waveforms cursors on the waveform plot – the white cursors with outward facing handles. The middle pair of arrow buttons, the single `<` and `>` symbols, start slow motion movement of the articulatory display in the forward and backward directions. Clicking on one of these buttons a second time or clicking on the other arrow buttons stops the slow motion movement. The inner arrow buttons move the articulatory display one time step in the backward and forward directions.

The `Drag` checkbox positions the articulatory position to the nearest 6.866 ms time step to a selected waveform cursor. The buttons with the white outward-facing handles reposition the waveform cursor to the displayed articulatory position. The left-facing handle button positions the initial waveform cursor while the right-facing handle button positions the final waveform cursor. That way you can find the nearest articulation to a point on the waveform display or you can locate the waveform sample matching a particular articulation.

The `Mark` button allows placing static markers on the x-y display, and this feature is used for marking palate points or maximum excursions of tongue points. Select `Mark`, select `Open` from the dialog to designate a point list file where the markers go, and select `OK`. Click with the left mouse cursor on the x-y display to place markers.

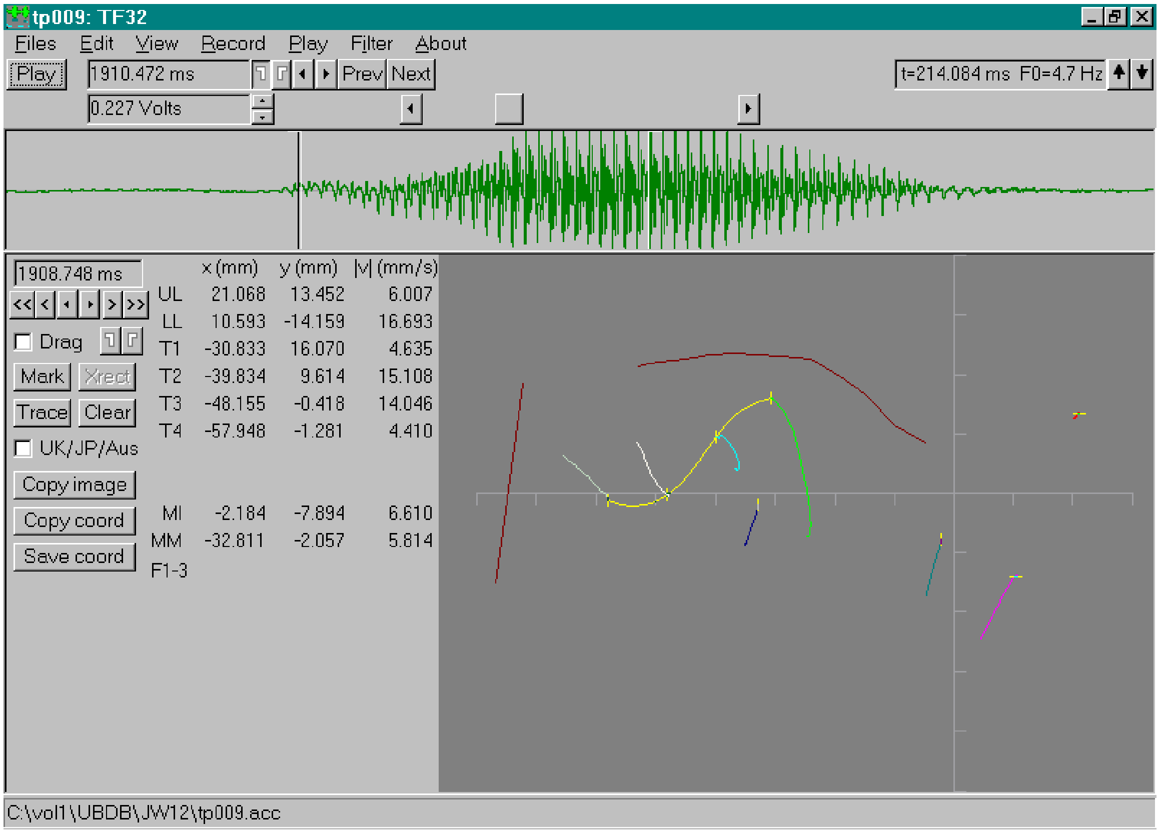
Select `Mark` to reactivate the dialog, and operate the `point` clicker to select a point you need to correct. That point will appear in red on the x-y display. Operate the x or y position clickers to move that point to make corrections. Select the `Delete` position if you want to remove that point. Select `OK` to save the changes.

The `Trace` button plots colored "glow worms" of articulatory trajectories for each pellet position for the entire interval selected with the waveform cursors while `Clear` removes those glow worms. The `Trace` feature is useful in conjunction with `Mark` for measuring the extents of "articulatory clouds" of pellet motion over long time intervals.



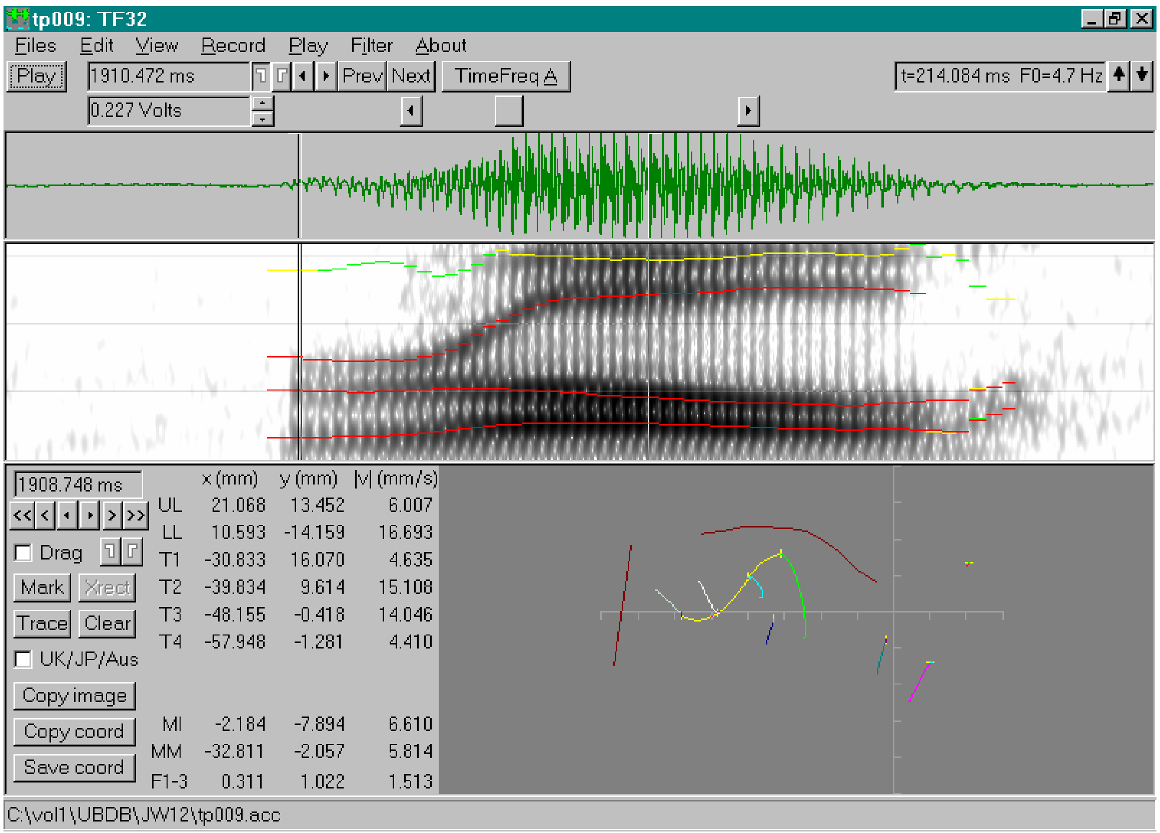
This figure shows the pellet marker clouds for subject JW12, record TP078, the initial portion of the Hunter Passage.

The `Trace` feature can also be applied to short intervals to help visualize articulatory gestures.

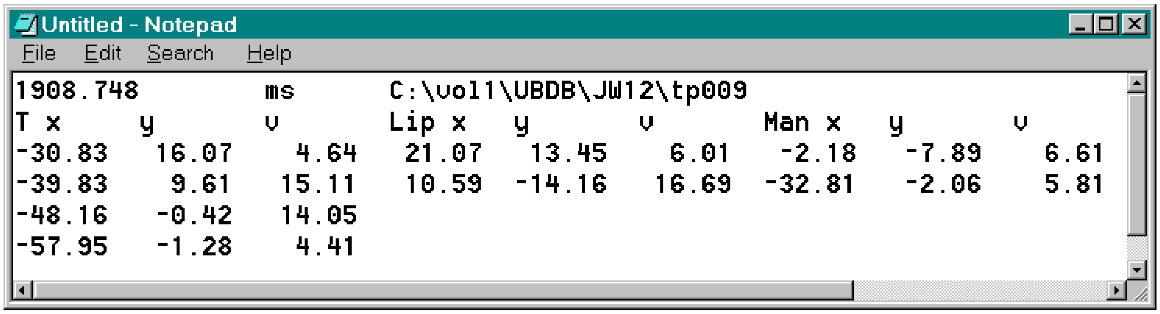


The figure shows the gesture for "row" for subject JW12, record TP009, where the gesture is plotted from the start of /r/ to the middle of the vowel as marked by the white cursors on the waveform plot with outward-facing handles, and where the tongue is located at the start of that gesture.

The `Copy image` button saves the x-y plot to the Windows clipboard for pasting as a figure into a word processor document. The `Copy coord` button copies the numeric coordinate readout to the Clipboard for pasting into a document or spreadsheet. The `Save coord` saves the numeric readout to a file. Selecting `View Open TimeFreq` from the TF32 menu, selecting the `LPC` checkbox on the time-frequency dialog to activate the formant overlay, and selecting OK produces the display



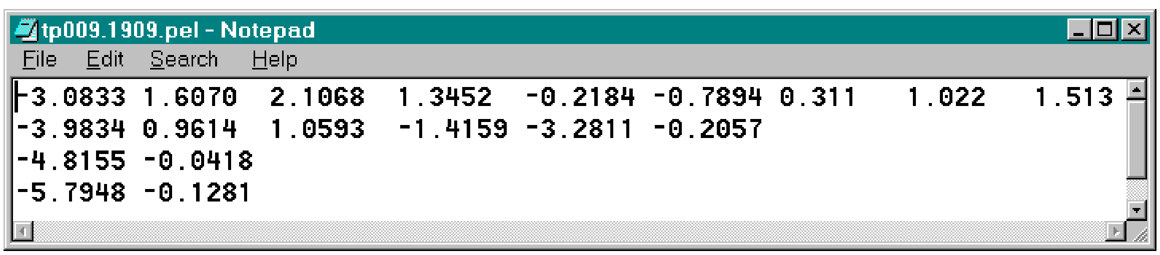
Selecting `Copy coord` from the x-y display control panel and pasting into Notepad results in




```
1908.748        ms       C:\vol1\UBDB\JW12\tp009
T x      y        v        Lip x    y        v        Man x    y        v
-30.83   16.07    4.64    21.07    13.45    6.01    -2.18    -7.89    6.61
-39.83    9.61   15.11    10.59   -14.16   16.69   -32.81    -2.06    5.81
-48.16   -0.42   14.05
-57.95   -1.28    4.41
```

The coordinates are in mm and follow the same layout as the coordinate readout on the x-y display. The formant frequencies are in kHz. Note that the x-y display time readout, which corresponds with the position of the black play cursor on both the waveform and time-frequency spectrogram plots, is listed with the coordinates.

Selecting `Save coord` from the x-y display control panel and viewing the saved ASCII file with Notepad shows




```
-3.0833 1.6070  2.1068  1.3452  -0.2184 -0.7894 0.311   1.022   1.513
-3.9834 0.9614  1.0593  -1.4159 -3.2811 -0.2057
-4.8155 -0.0418
-5.7948 -0.1281
```

While the time stamp does not appear in the file, it is part of the default file name for saving coordinates. The coordinates are in cm for compatibility with articulatory modelling software that uses these data. The first two columns are x and y coordinates of tongue points T1-T4. The next two columns are upper lip followed by lower lip, the two columns after that are lower jaw markers MXI and MXM (incisor and molar), and the last three columns of the first line are formant frequencies F1 through F3 in kHz.

5.7 Time-slice spectrum

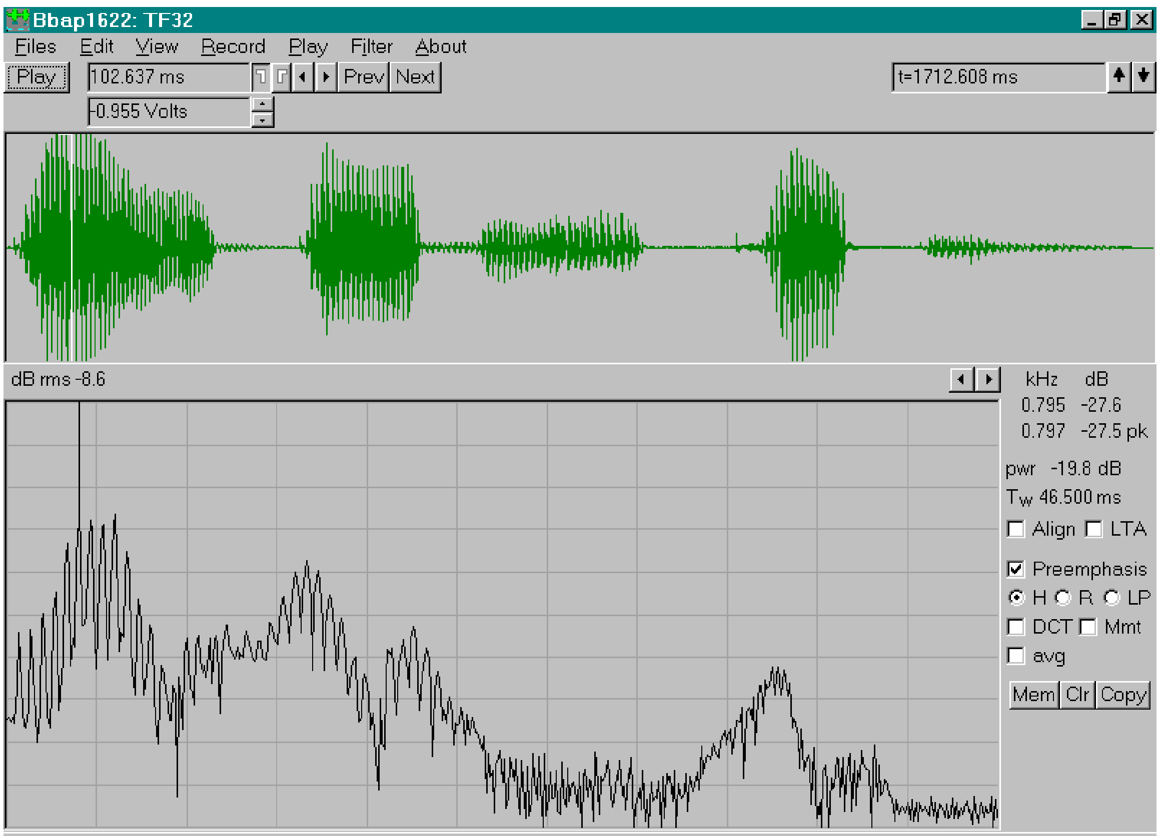
Select `View Open Spec` from the main menu to display the time-slice spectrum of the selected waveform channel. When the TF32 main window is maximized, the time-slice spectrum window is attached to the main window as seen on the next page. Select `View Spec` from the main menu to close the spectrum display. When the TF32 main window is not maximized or when the x-y articulatory display is open, the time-slice spectrum appears in a separate overlapping window. In this instance, select `View Spec` to bring the spectrum display to the foreground if it was hidden behind another window. Check the `X` button on the right-hand spectrum window title bar to close the spectrum window when it is displayed as an overlapping window.

The time-slice spectrum displays a single spectrum computed from a cursor-selected interval of the selected waveform channel. When the `Align` button is unchecked, the spectrum window (waveform time-slice used to compute the spectrum) is the cursor-selected interval, limited to a maximum of 1024 samples. When this limit applies, the spectrum window starts at the left cursor. The T_w readout shows the length of the selected spectrum window in ms. When the `Align` button is checked, buttons appear with the left and right handle cursor icons: use these buttons to center the spectrum window on either the left or the right waveform cursor. A clicker appears next to the T_w window length readout: use this clicker to change the length of the spectrum window when `Align` is checked.

The spectrum is automatically recomputed when waveform cursors move or when the waveform changes in response to opening a waveform file or filtering a waveform channel. For a real-time display of the time-slice spectrum during recording, check `Align` and select the right handle-cursor button – that way the spectrum will be computed from the part of the recorded waveform that scrolls in from the right-hand edge of the waveform display.

Check the `LTA` button to compute a long-term average spectrum. Starting with a spectrum window whose left edge is aligned with the initial cursor, this mode averages the magnitude-squared Fourier spectrum for successive windows with 50 percent overlap. The T_w readout shows the length of each of these windows (in ms), and a clicker allows changing this value. The long-term average spectrum is automatically computed when you move the waveform cursors, but this computation can take several seconds if a long interval is selected. The spectrum plot will show gray when this computation is in progress and will show black when the computation is complete. While recomputing a long-term average spectrum will not impede cursors or screen scrolls, the long-term average spectrum will not be updated in real time during a screen scroll.

Click on the left-right arrow buttons on the upper right of the spectrum display or drag the left mouse cursor over the spectrum plot to set the frequency cursor and get a readout of the spectrum as seen below. The arrow buttons are particularly useful for fine control over the frequency cursor. Activating the frequency cursor makes `kHz` and `dB` readouts appear on the upper part of the spectrum control panel. A pip will also appear on the time-frequency spectrogram analysing the same waveform; the time extent of that pip will show the placement of the time-slice spectrum window while the frequency location (y-coordinate) of that pip will match the time-slice spectrum frequency cursor. You may need to select a frequency on the time-slice spectrum low enough for the pip to appear on the displayed frequency range of the time-frequency spectrogram.



The first line of the spectrum frequency cursor readout gives the kHz and dB values of the computed spectrum at the cursor location. The second line gives the kHz and dB values of the last spectrum peak you had scanned across with the frequency cursor. Parabolic interpolation is used to refine the location of the peak frequency as well as the dB value of the peak. The frequency location has to be placed at a peak for that readout to update; you may need to use the left-right arrow buttons to position the frequency cursor accurately enough to find the peaks you want.

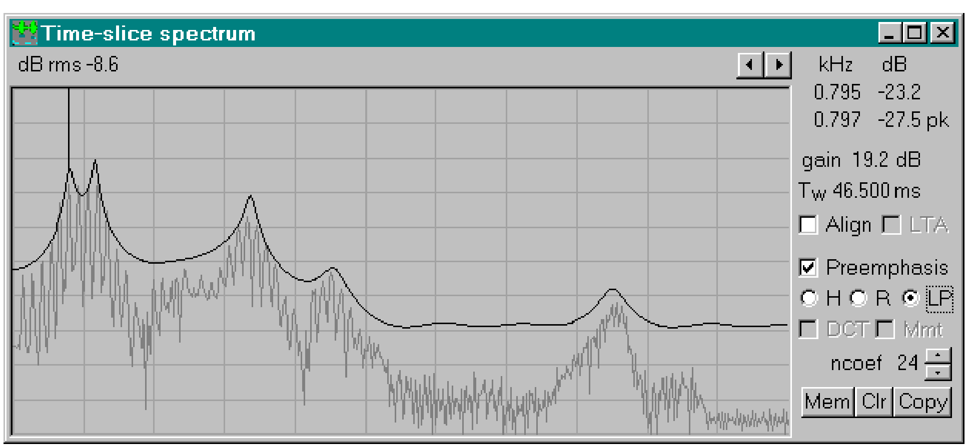
When the TF32 main window is maximized and the time-slice spectrum window is attached to the main window, the `Edit Copy readouts to clipboard` and `Edit Measure sequence` menu commands will append numeric readouts from the time-slice spectrum to the numeric readouts from the waveform and analyser displays. Be sure the TF32 main window is maximized and the time-slice spectrum window is attached if you want to save numeric values from the spectrum window in this way.

Checking `Preemphasis` computes the first-difference $d[n] = s[n] - s[n-1]$ from waveform samples $s[n]$. First-difference waveform $d[n]$ is used to compute the spectrum when `Preemphasis` is checked. Unchecking `Preemphasis` computes the spectrum directly from the waveform samples $s[n]$. The time-frequency spectrogram also has a preemphasis selection. The effect of preemphasis is to increase the dB level of higher spectrum frequencies in the amount of 6 dB for each octave (doubling of frequency). Preemphasis is commonly used in the analysis of speech waveforms and is the default selection for TF32.

The lower-middle section of the spectrum control panel contains a cluster of buttons for choosing between the Fourier spectrum, Hamming or rectangular window (`H` and `R` buttons), and LPC spectrum (`LP` button). A Fourier spectrum may be post-processed by the discrete cosine transform smoothing function (`DCT` button), the moments model spectrum (`mmT` button), or by averaging the magnitude-squared spectrum in 1 kHz bands (`avg` button). These modes are described in the sections that follow.

The bottom portion of the spectrum control panel contains the buttons `Mem`, `Clr` and `Copy`. The `Mem` button (as in memory) saves the last computed spectrum as a gray trace – this allows comparison with any newly computed spectrum that appears as a black trace. The `Clr` button clears that selection, showing only a single spectrum plot. The `Copy` button copies the spectrum plot, both gray and black traces, to the Windows clipboard for pasting figures into documents.

For example, selecting `Mem` to save to Fourier spectrum as a gray trace and selecting `LPC` to compute the LPC spectrum and plot it as a black trace produces



5.7.1 Hamming-window Fourier spectrum

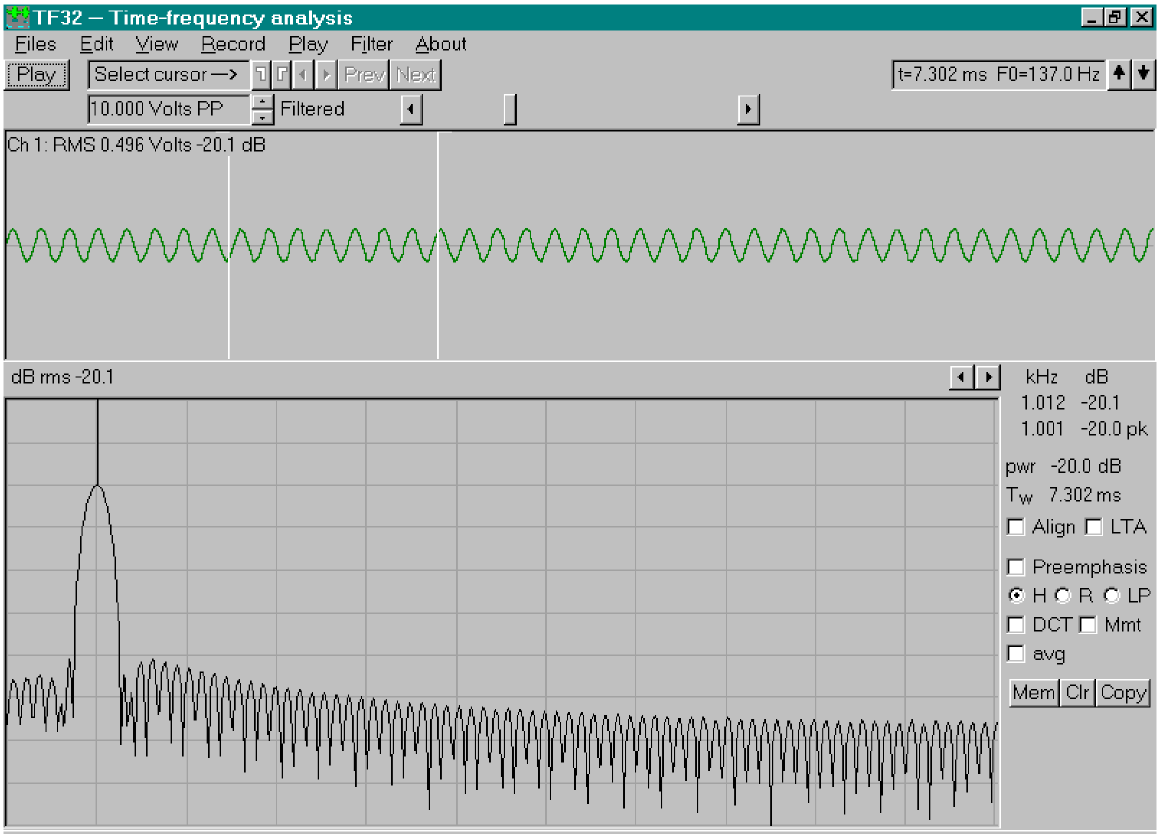
The Hamming window Fourier spectrum (the `H` button) multiplies the selected waveform interval by the function $0.54 - 0.46\cos(2\pi n/N)$, pads the windowed interval with zeroes to a length of 1024 samples, and takes the Fourier transform. The frequency interval from 0 to 1/2 the sampling frequency is plotted with 512 divisions.

This mode is useful for measuring periodic signals such as voiced speech waveforms or other waveform composed of one or more sinusoids. The Hamming window gives good dynamic range in measuring sinusoids that vary greatly in magnitude; a rectangular window produces strong spectrum side lobes where a strong sinusoid can mask out a weak sinusoid, even if it is far away in frequency.

The Hamming window in TF32 is scaled to facilitate dB measurements of sinusoidal signals. The dB scale in TF32 reads 0 dB for a sinusoid of RMS amplitude half the allowed peak amplitude of the waveform channel. In more concrete terms, a default channel range is 20 Volts peak-to-peak or 10 Volts peak. So a sine wave with RMS amplitude 5 Volts (peak amplitude $5\sqrt{2} = 7.07$ Volts) is the largest amplitude sine wave in round numbers that will fit in a channel – this is the 0 dB reference level. A sine wave of peak amplitude 0.707 (RMS value 0.5) is then -20 dB, or a factor or 10 in magnitude, below the reference level.

The figure that follows is generated by these steps. Invoke `Files New` from the main menu to get a zero-valued waveform channel, invoke `Filter Sine` and enter 1000.0 into the frequency edit box, 0.707 into the magnitude edit box, and select `OK`, use the waveform cursors to select a 7.3 ms interval, activate the dB waveform readout by invoking `View Wave plot` and by selecting `RMS dB` from the dialog, activate the spectrum display, uncheck `Preemphasis`, and maximize the TF32 main window.

For this example, make sure that `Preemphasis` is unchecked so that the spectrum display gives similar dB readouts to the waveform channel dB readout.



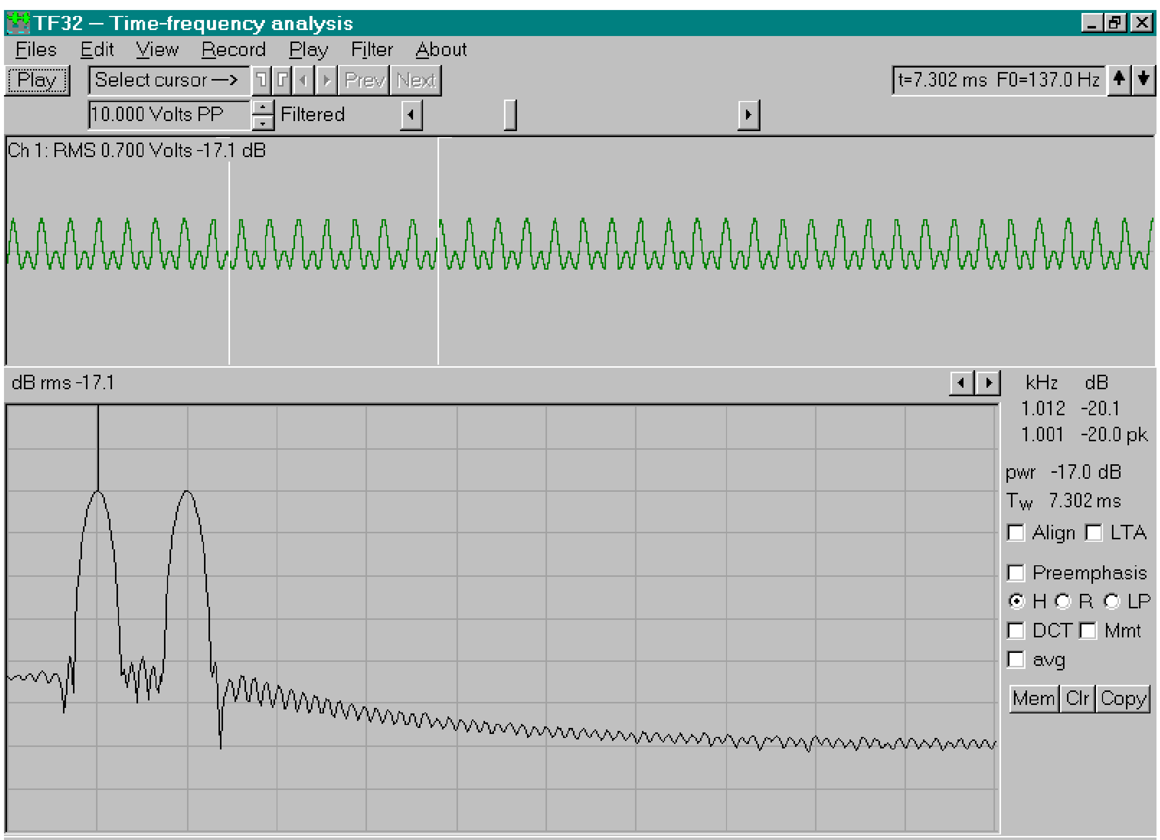
The waveform channel dB readout is for the entire cursor-selected interval: it reads an RMS value of 0.497 Volts and -20.1 dB on account of edge effects in positioning the waveform cursors relative to sine wave cycles. Positioning the cursors on an integer number of cycles should give readings of 0.5 Volts and -20 dB. The `dB rms` readout on the upper left of the spectrum display is the dB level of the spectrum window interval before any preemphasis or windowing used to calculate the spectrum. Since the cursor selected interval is at or below 1024 sample (46.5 ms in this example using a 22.05 kHz sampling rate), the waveform channel and spectrum `dB rms` readouts show the same values.

The `pwr` readout on spectrum control panel to the right of the spectrum plot shows -20.0 dB. This readout is called “power” because the level of a sinusoid relates to signal power; it is only meaningful to talk about power because the energy level of a steady-state sinusoid is infinite. This readout is computed *after* any selected preemphasis or spectrum window; it can show a different number than the waveform dB readout. In this example, preemphasis is turned off but a Hamming window is used – the readout of -20.0 dB is actually more accurate because the Hamming window tapers off at the edges, mitigating edge effects, both for dB level and spectrum calculations.

A spectrum plot of one or more sinusoids analysed with a Hamming window results in one or more "lumps." The spectrum in the example has a single lump for the one sinusoid. Each lump is the frequency response of the Hamming window itself. The dB span from the top of the lump to the tops of the ripples to the sides of the lumps is a property of the particular window taper. The Hamming window spans 42 dB. A neighboring sinusoid more than 42 dB below the main sinusoid will be masked out by these "side lobes" and escape detection. The width of each of the lumps is controlled by length of the window; a longer window gives a narrower lump, allowing sinusoids closer in frequency to be resolved.

Notice that the interpolated spectrum peak readout in the upper right of the spectrum display (top of the spectrum control panel) reads 1 kHz (1000 Hz) and -20 dB, the frequency and dB level of the waveform itself.

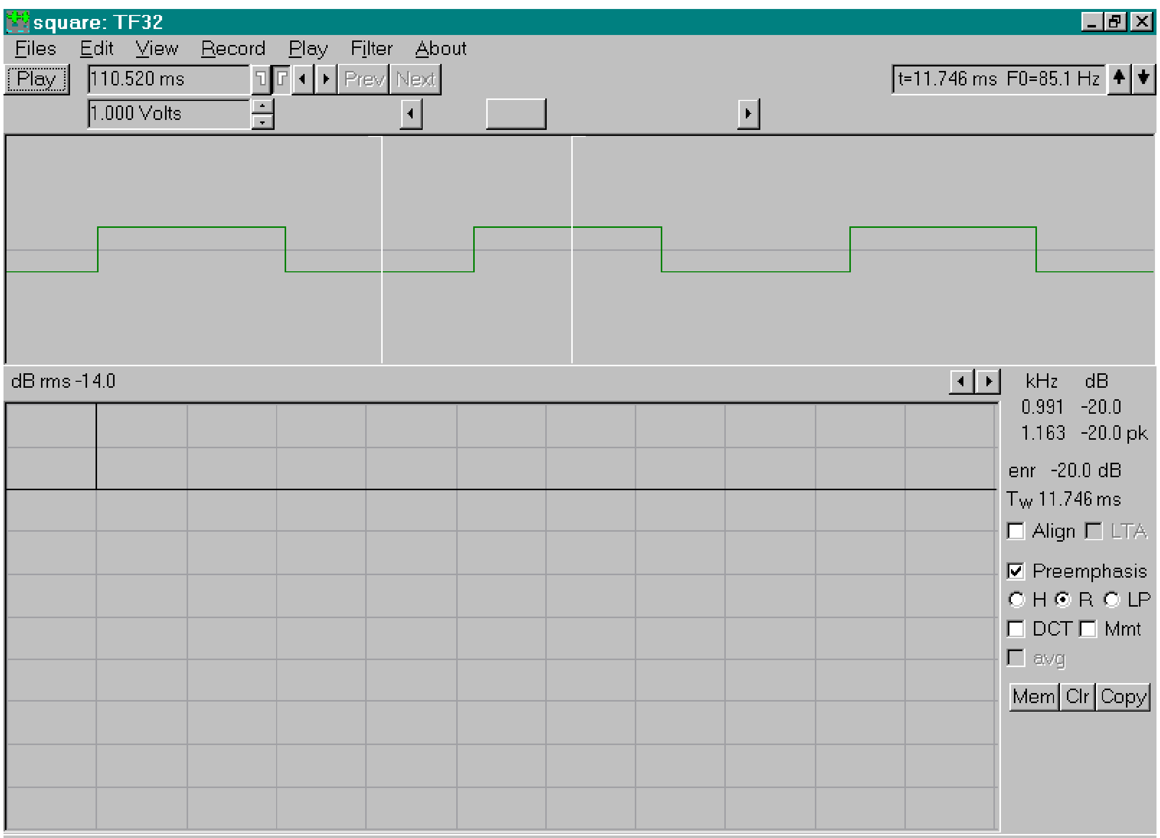
Adding a second sinusoid, of equal amplitude and at 2 kHz increase the signal level by 3 dB to -17 dB, but the peak of each sinusoid remains at -20 dB. The reason for this is that the power levels of sine waves of two different frequencies add according to Parseval's Theorem of Fourier Analysis and doubling power increase the dB reading by 3 dB.



5.7.2 Rectangular-window Fourier spectrum

A sine wave or a signal that is a sum of sine waves is infinite in duration; a Hamming window is an effective way of truncating such a signal so we can compute the FFT to generate a spectrum and interpret the results. A finite-duration pulse waveform is already finite-duration – we don't need to apply a tapered window to truncate it. Applying a rectangular window long enough to enclose the pulse waveform implies "zero padding" the signal outside the window, which does not change anything because the signal is already zero outside the window.

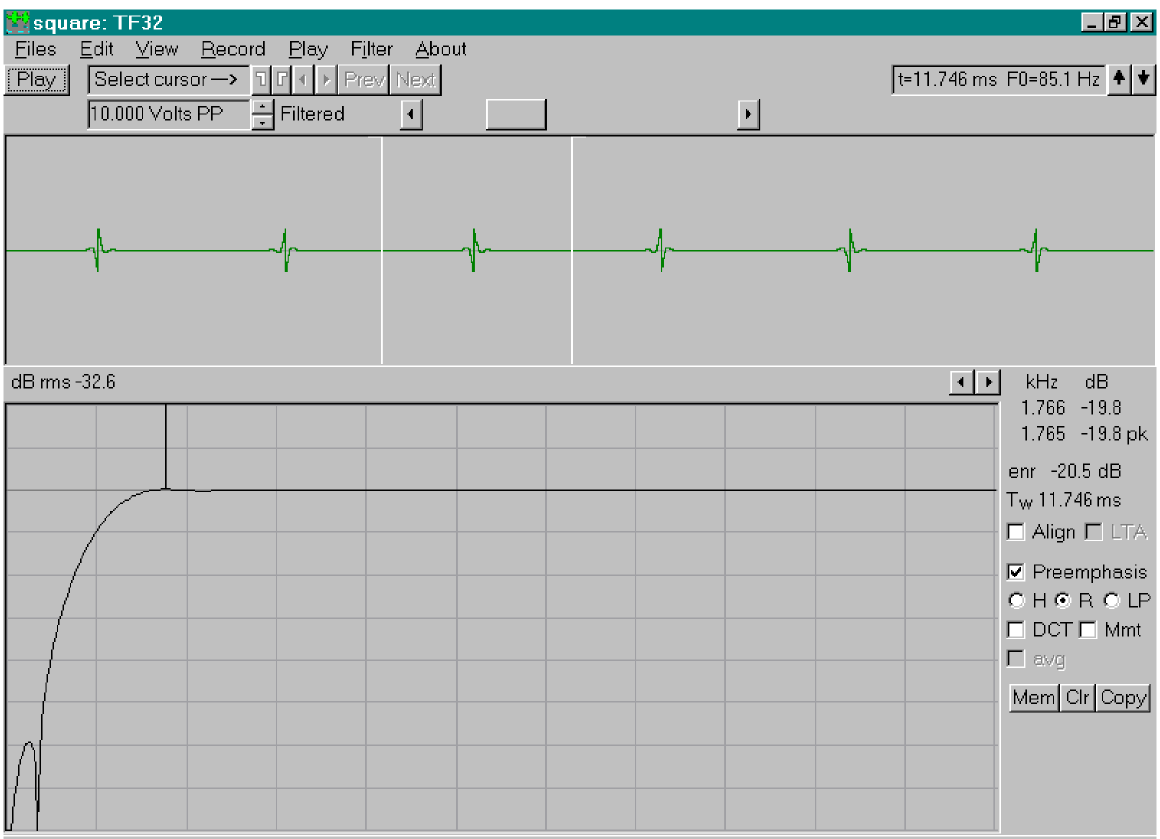
The rectangular window setting (`R` button) can compute the Fourier spectrum of a pulse waveform such as the impulse response of a filter and is useful for measuring the frequency response of a filter from its impulse response. Opening a waveform file containing a square wave of alternating 1 and -1 volt levels, positioning the cursors to bracket one square wave transition between 1 and -1 volts, checking the `Preemphasis` box, and selecting the `R` button results in



Checking `Preemphasis` is important because it differences the square wave to convert it to an alternating impulse train inside the spectrum analyser. Each impulse has a flat spectrum. A square wave of peak amplitude 1 is -20 dB; a square wave of peak amplitude 10 (maximum allowed signal for a peak-to-peak range of 20 Volts) is 0 dB.

Note that the control panel readout is now labeled **enr**, it reads -20 dB, and it differs from the -14 dB RMS reading on both the waveform and spectrum displays. This denotes energy of the pulse waveform enclosed by the rectangular window, and the **enr** value will remain constant provided the waveform cursors select the entire extent of a single pulse, even if you move the cursors. The other two dB readouts are of **power**, and the values will change when you move the cursors because energy is power divided by time. Notice that a constant -20 dB spectrum gives an energy reading of -20 dB.

To produce the following plot, select **Filter Hipass** from the TF32 main menu, enter 1000 Hz and select OK. This action filters the square wave and changes the spectrum plot to show the frequency characteristic of the filter. While the high-pass filter default setting of 50 Hz is best for preconditioning audio waveforms, the value of 1000 Hz in this example better shows the frequency response features on the spectrum plot.



Exciting the filter with a square step and then applying preemphasis is effectively the same as exciting the filter with impulses and not using preemphasis. The advantage of testing a filter by exciting it with a square wave followed by preemphasis is that digital filters are not perfectly linear; roundoff error can make the filter output stick at a small contant level, and this procedure cancels out that constant level.

Note that the `enr` readout of the filtered waveform is -20.5, slightly below -20 db; this is consistent with filtering out only a small chunk of the spectrum. Filtering out half of the spectrum would bring `enr` down to -23 dB because each reduction of 1/2 in energy is -3 dB. Also note that the flat part of the spectrum stays at -20 dB, but there was a .2 dB rise (1.765 kHz, peak at -19.8 dB) past the corner frequency.

5.7.3 LPC spectrum

Linear-predictive coding (LPC) analysis determines the coefficients of a causal all-zero inverse filter with leading coefficient of one that minimizes the sum of squares of the filtered waveform over a selected interval. This has been shown to be equivalent to flattening the spectrum of the filtered waveform, and the all-pole forward filter to the least squares-minimizing spectrum-flattening inverse filter has a frequency response that approximates the original signal spectrum. Because the vocal tract filter for vowels is all-pole and because even a pole-zero filter as arises with nasals can be approximated with an all-pole filter, the frequency response of the LPC all-pole filter provides an estimate of the vocal tract filter frequency response and hence the formant structure of the speech waveform.

Put another way, starting with work by Atal, Makhoul, Markel, Wakita, Childers and others in the early 1970's that involved numerous simplifying assumptions that we knew then and know now are not quite accurate, LPC is a technique for designing a filter matching the speech spectrum that produces formant estimates of questionable reliability, but in 30 years, no one has come up with anything that is any better.

TF32 implements LPC analysis using the covariance method that calculates a least-squares inverse filter output signal over the speech signal interval selected with the waveform cursors as selected by the `Align` check box. A constant term of -60 dB in level compared to the signal (factor of 1000 in magnitude-squared) is added to the covariance matrix to prevent ill-conditioned matrix inversion: this has the effect of adding "fill" to the lows of the LPC spectrum plot. Selecting `LP` from the spectrum control panel activates a clicker for changing the number of coefficients from the default value of 1 coefficient per kHz sampling rate plus 2 more for voice source and nasalization effects. The LPC spectrum plot is scaled so that the LPC spectrum skims the tops of the harmonics on the Hamming-window Fourier spectrum at a fundamental frequency of 140 Hz.

Instead of reporting `pwr` or `enr`, the `LP` selection reports prediction gain in db as the readout labelled `gain`. Prediction gain is the amount that the least squares signal is lower in energy compared with the original speech waveform, and it is a measure of the degree of spectrum flattening. A high prediction gain means a very peaky LPC spectrum. Enabling preemphasis will reduce prediction gain as the spectrum will have a reduced range between highs and lows.

Fourier spectrum post-processing

The checkboxes `DCT`, `Mmt`, and `avg` select spectrum plot post-processing – they only apply to the Fourier spectrum and are disabled when the LPC spectrum is selected. Each of these post-processing modes operates on the Fourier spectrum to produce a modified spectrum plot along with numeric parameters. The resulting numeric parameters (11 DCT coefficients for `DCT`, 4 moment parameters for `Mmt`, or dB values for 1 kHz bands for `avg`) are saved by the `Edit Copy readouts to clipboard` and `Edit Measure sequence` commands of the TF32 main menu. The numeric parameters are added to the numeric readouts extracted from the waveform plots and analysers when the TF32 main window is maximized and the time-slice spectrum is attached to the main window. When the time-slice spectrum is displayed as a separate overlapping window, numeric values from the time-slice spectrum are not saved by the `Edit` commands from the main menu – make sure you have selected the required mode.

5.7.4 Discrete cosine smoothed spectrum

The `DCT` selection smooths the Fourier spectrum by applying an 11-coefficient (DC plus 10 frequency terms) discrete-cosine transformation (DCT) to the frequency band from 0 to 1/2 the spectrum range (1/4 the sampling frequency). The smoothing is applied to spectrum dB values and each coefficient value gives the peak spectrum deviation for that DCT term in dB. This measure is motivated by S. A. Zahorian, and A. J. Jagharghi (1993), Spectral-shape features versus formants as acoustic corrleates for vowels, JASA 94 1966-1982. The 11 DCT numeric values are saved by the `Edit Copy readouts to clipboard` and `Edit Measure sequence` commands of the TF32 main menu when the main window is maximized and the time-slice spectrum window is attached.

5.7.5 Moments-model spectrum

Spectrum moments, as described by Forrest, K., Weismer, G., Milenkovic, P., Dougall, R. N. (1988). Statistical analysis of word-initial voiceless obstruents: preliminary data. Journal of the Acoustical Society of America 84, 115-123, is a method for turning a spectrum shape into a set of numbers. The Fourier power spectrum, before computing dB, is treated as if it were a probability density function, and the method computes the mean and standard deviation (units are kHz) as well as dimensionless coefficients skewness and kurtosis of the higher moments. Spectrum moments can be used for unimodal spectrum shapes as arise in fricatives; they are less successful in characterizing vowel spectra as these spectra tend to be multi-modal.

Spectrum moments turn a spectrum shape into numbers; a method for turning those numbers back into a spectrum shape is described by Milenkovic, P.H. (1999). Analysis and synthesis of unvoiced speech using spectrum moments, Workshop on spectrum moment measures for speech language and hearing research, ASHA, San Francisco. A description of the algorithm is given by http://www.medsch.wisc.edu/ milenkvc/tools.html. This method is used to produce the spectrum plot selected with the `mmT` checkbox. Selecting `mmT` calculates spectrum moments, reports moments values on the top of the spectrum display, and calculates a smooth spectrum shape matching those moment values. This spectrum calculation works best on Fourier spectra that are uni-modal.

A set of four numeric values (mean, standard deviation in kHz, dimensionless skewnew and kurtosis) are saved by the `Edit Copy readouts to clipboard` and `Edit Measure sequence` commands of the TF32 main menu when the main window is maximized and the time-slice spectrum window is attached.

5.7.6 Average spectrum in 1 kHz bands

Selecting the `avg` checkbox computes the average magnitude-squared Fourier spectrum in 1 kHz bands and plots the results in dB. The initial band starts at 100 Hz and ends at 1 kHz to remove low-frequency recording artifact from the measurement. Subsequent bands are 1 kHz wide until the last band that ends at 1/2 the sampling frequency. The spectrum plot shows a square-step spectrum – moving the frequency cursor to each square step to reads out the dB signal power for that band.

On account of Parseval's theorem relating the summed magnitude-squared Fourier spectrum to power in the signal, this analysis mode can be thought of as a bank of band-pass filters, and the dB values are the signal power levels in each filter band. For a signal with power concentrated in one band and with the preemphasis turned off, the dB reading for that band will be the same as the RMS dB reading for that waveform channel, where an RMS reading of 1/2 the maximum signal value for that channel is 0 dB. If the signal power is split equally between two bands, each band will read 3 dB less than the dB value for the signal on account of a factor 1/2 in power being -3 dB. There is not a simple relationship for the signal split between multiple bands on account of the dB values for the bands not adding up to the dB value for the signal.

The dB values for the 1 kHz bands are saved by the `Edit Copy readouts to clipboard` and `Edit Measure sequence` commands of the TF32 main menu when the main window is maximized and the time-slice spectrum window is attached.

5.8 Jitter-shimmer-voice aperiodicity SNR

Jitter is the cycle-to-cycle variation in the pitch period during voicing; shimmer is the variation in amplitude between waveform cycles; periodicity signal-to-noise ratio compares the magnitude of a voiced signal to the magnitude of the aperiodic componenet. While these measures are coupled – a synthetic signal with just one measure will show leakage into the other two measures with any practical measurement algorithm – the profile of all three measures has been employed in research studies to characterize the voice.

These aperiodicity measures are a direct reimplementation of the algorithm used in the CSpeech software. As these particular measures have found such widespread use and are unique to CSpeech, they are implemented without any improvements: that way comparisons can be made with earlier studies. The analysis is applied to the *unfiltered* waveform to be consistent with CSpeech. When analysing a waveform recording that has not been filtered, consider applying the `Hipass` filter from the `Filter` menu to remove any low-frequency artifact from the signal.

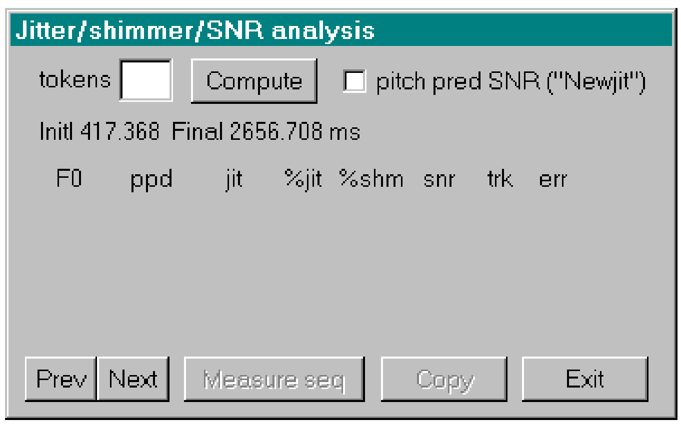
The voice aperiodicity measures use the algorithm of Milenkovic, P. (1987), Least mean square measures of voice perturbation. Journal of Speech and Hearing Research 30, 529-538. The algorithm determines an initial pitch period using the autocorrelation method with center clipping. It then moves a pitch-period long window across the waveform and adapts that window length to changes in the measured pitch period. The waveform from that pitch-period long window is compared in a least-squares sense with the waveform from one pitch period earlier by computing the cross-correlation function. The parabolically-interpolated peak of the cross-correlation function for different time lags around the estimated pitch period is used to determine the instantaneous pitch period to resolutions better than the time interval between waveform samples, and the difference in the instantaneous pitch period sample a pitch period apart gives instantaneous jitter. The parabolically interpolated correlation peak is also used to calculate the amplitude variation or shimmer as well as the aperiodicity SNR.

An alternate measurement of the aperiodicity SNR (jitter and shimmer are the same) is given by Milenkovic, P. H., Bless, D. M., and Rammage, L. A. (1991), Acoustic and Perceptual Characterization of Vocal Nodules. In Vocal Fold Physiology: Acoustic, Perceptual, and Physiological Aspects of Voice Mechanisms, J. Gauffin, and B. Hammarberg editors, Singular Publishing Group, San Diego, 265-272. This algorithm is the `Newjit` menu command in CSpeech; it is the `Pitch pred SNR ("Newjit")` checkbox in the Jitter/shimmer/SNR analysis dialog box selected with the `View Open Jitter` menu command.

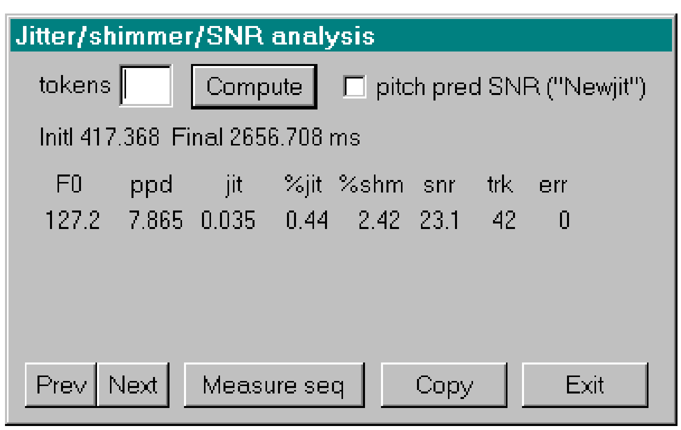
The change is to measure SNR by subtracting two waveforms separated one pitch period apart to directly measure the aperiodic component rather than use an indirect measure based on the strength of the cross-correlation peak. This subtraction requires shifting one of the two waveforms by a pitch period value that can be a fractional number of waveform samples in length, which is done using a sample interpolating filter. This newer method tends to report higher SNR values (lower levels of noise) on account of using a shorter sliding waveform window for the SNR calculation (1.5 ms), which will naturally bias the results to higher SNR. Think of the older SNR as a lower bound on the true SNR and the newer SNR as an upper bound.

To apply the voice aperiodicity analysis, select a voiced interval with the waveform cursors. If the signal has not already been filters, apply `Highpass` from the `Filter` menu before analysis to remove low-frequency artifact. The initial cursor needs to be in a voiced portion of the speech waveform to reliably determine the initial pitch period. The algorithm will then track changes in the pitch period from that initial value, so it is important that voicing continues over the entire cursor-selected interval.

Selecting `View Open Jitter` from the TF32 menu activates the dialog

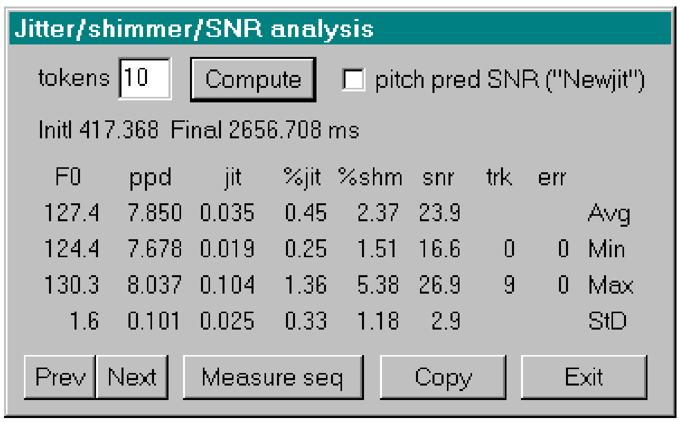


Note that the selected waveform interval ranges from 417 to 2656 ms. One option is to analyze the entire interval as one uninterrupted block. To do this, leave the `tokens` edit box blank and select `Compute` to produce the following display



F0 is in Hz, pitch period in ms, (absolute) jitter is in ms, percent jitter and shimmer are fractions expressed as percent, and SNR is in dB. The numbers `trk` and `err` are reliability measures for the pitch tracker. A high `trk` count means large swings in F0 while a high `err` count indicates voice breaks that disrupt the pitch track.

Another option is to divide the long voiced interval into 100 ms-long tokens, equally spaced in the selected interval. Enter the number `10` in the edit box: this action grays out the numeric readouts, indicating that you need to select `Compute` to update them. Selecting `Compute` produces



The use of multiple tokens allows computing statistics which indicate the variability of the voice perturbation measures over a long sustained voicing. For shorter voiced intervals, such as selections from speech context syllables, there is not enough signal duration to segment into tokens – leaving the `tokens` edit field blank is recommended for those cases.

Since the voice periodicity measures appear in a dialog box, that dialog box has `Prev` and `Next` buttons for selecting the previous and next waveforms in a sequence, a `Measure seq` button for measuring all the waveforms in a sequence and saving numeric results to a file, and a `Copy` button to copy one measurement at a time to the clipboard. When using `Prev`, `Next`, or `Measure seq` to select waveforms from a sequence, it is important that the waveforms in that sequence have been selected to only include voiced intervals – otherwise the jitter analysis will give spurious results.

To use the `Measure seq` button to do batch mode analysis, it is best to cursor off voiced intervals of waveforms of interest and to add those intervals as entries to a wave list file with the `Edit Add index to wave list` command. This technique restricts the analysis to voiced intervals without having to save a larger number of waveform files containing voiced snippets. After adding the required entries, open that wave list with `Files Open`, activate the Jitter dialog with `View Open Jitter`, select the desired number of tokens, if any, check or uncheck the `pitch pred SNR ("Newjit")` box, and select `Measure seq` to initiate batch-mode measurement.

Reliablity of pitch estimates

The validity of all the voice perturbation measurements depend on the accuracy of pitch measurements, and unfortunately, no pitch measurement method is without error. The analysis depends on a correct initial pitch estimate made by the autocorrelation method, and the analysis depends on the ability to track the small variations in pitch produced during sustained voicing. Even if the initial pitch estimate is valid, breaks in the voice can interrupt the pitch track that follows. As with any automatic analysis applied to speech, it is important to interpret numeric results and make judgements regarding analysis artifact based on the interpretation of those results.

The most common type of error you will encounter is pitch period doubling. Instead of the true pitch period of 4 ms (250 Hz), the analysis may report a pitch period of 8 ms (125 Hz). This type of pitch period doubling will occur when there is "double pulsing" of the voice waveform. Instead of a regular oscillation at 250 Hz, one will observe an oscillation that is somewhat irregular at 250 Hz but is regular when regarded as an oscillation of 125 Hz. We believe that in most cases of pitch period doubling, you will see some evidence of such "double pulsing" if you observe the waveform closely enough. Monitoring the reported pitch period can help spot that condition.

The next most common error is to erroneously identify the first formant frequency as the pitch period. This will be evidenced by F_0 values that are much too high for the speaker in question.

There are two actions you can take when discovering pitch error. One action is to move the starting cursor of the voice analysis to find an interval of the waveform that gives a corrected starting pitch. You may have placed the initial cursor too close to the beginning of voicing, and moving the initial cursor further into the voiced segment may help.

The other action may be to accept the reported pitch as simply part of the way the analysis turns out. With pitch doubling, the true pitch period may give high levels of jitter and shimmer while the doubled pitch period may give normal levels of jitter and shimmer at an unusually low reported F_0. Each experimenter needs to decide what measurement is reasonable in such cases.

A more difficult condition to gauge is whether the pitch tracking is disrupted by voice breaks. You can check for unusually high jitter and shimmer, low SNR, and reported pitch period outside of norms for that subject. The `trk` and `err` counts can also help. The `trk` count is the number of instances that the pitch tracking search window had to be shifted to track a large deviation in the pitch period. A large `trk` count indicates large swings in pitch without any other problems with the measurement. The `err` count is the number of instances where two different measures of the pitch period disagree by more than one sample position. A large `err` count indicates breaks in voicing that may exagerate the jitter and shimmer values and diminish SNR for what is already unsteady voicing.

6 Aerodynamic measurement

The Lab Automation level of TF32 is configurable to conduct voice aerodynamic data acquisition along with the automatic calculation and logging of aerodynamic measures. The text file `aero.pre` contains keywords that specify settings for this task. TF32 may be installed so that icon labeled Aero on the Windows desktop starts TF32 and automatically opens the aero preset file when you double-click that icon.

The Aero system samples the audio signal at a 16 kHz rate, and it samples DC-coupled pressure and flow signals, each channel at a 2 kHz rate. Waveforms are automatically stored in a lossless data compression format; the retrieved waveform samples are identical to the original values. Sampling at 12 bit A/D resolution allows a recordable CD-ROM to store up to 22 hours of recordings; 10 gigabytes of disk space can store up to 340 hours of recordings. With this storage efficiency, the Aero system saves all waveforms recorded from your subjects for archival purposes – the raw waveforms do not have to be discarded on account of storage limitations.

The Aero system supports logging of waveform measurements during acquisition; it also allows for the post-acquisition viewing and analysis of the archived waveform recordings. The software is flexible in the selection of numeric waveform measures (cursor endpoints; min, max, mean, and std deviation waveform statistics, RMS and dB waveform values); the software offers pitch, waveform dB contour, and wideband and narrowband time-frequency spectrograms as analysis choices.

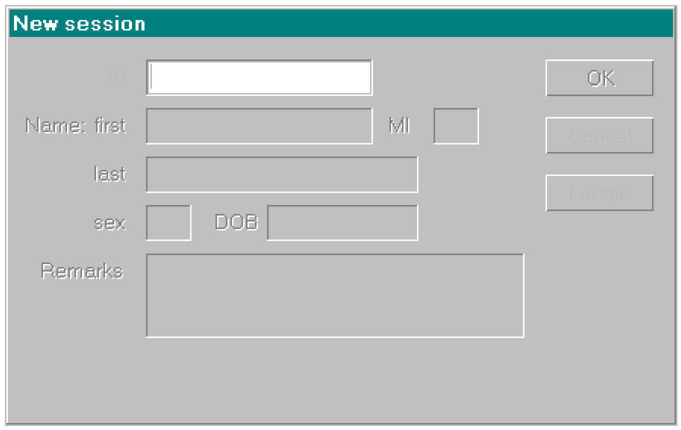
A smooth-scrolling display of signal waveforms along with a real-time update of analysis results gives the operator feedback during the waveform recording; these displays may be used for subject feedback.

The Aero system includes algorithms for the operator-assisted extraction of resistance from the pressure and flow traces. There are two resistance measures. Laryngeal resistance R_g is calculated from the alternating patterns of pressure and flow that occur in the production of consonant-vowel sequences (CV's). Velar port resistance R_v is calculated from the rise in air flow that is correlated with the rise in pressure. There are two calculations modes. In one mode the software automatically locates the alternating intervals of peak pressure and phonation flow, counts the number of CV's and computes average resistance for R_g; the software correlates the pressure and flow traces to compute R_v. In a second mode, the software computes R_g and R_v based on pressure and flow values at cursor locations; this mode requires the manual placement of cursors.

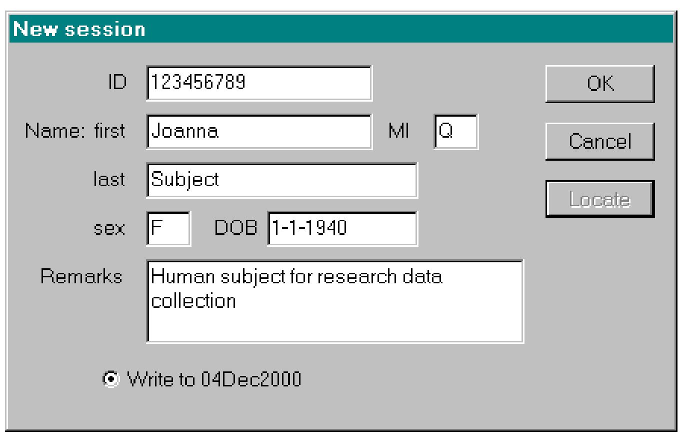
6.1 Procedures

6.1.1 Specifying subject information

Activate TF32 for aero analysis by double clicking the `Aero` icon on the Windows desktop. The initial screen displays the dialog



To check the operation of the software, enter the id number `123456789` and press the `Locate` button. You will get a blank fields because that subject is not yet in the system. Fill those fields to look like

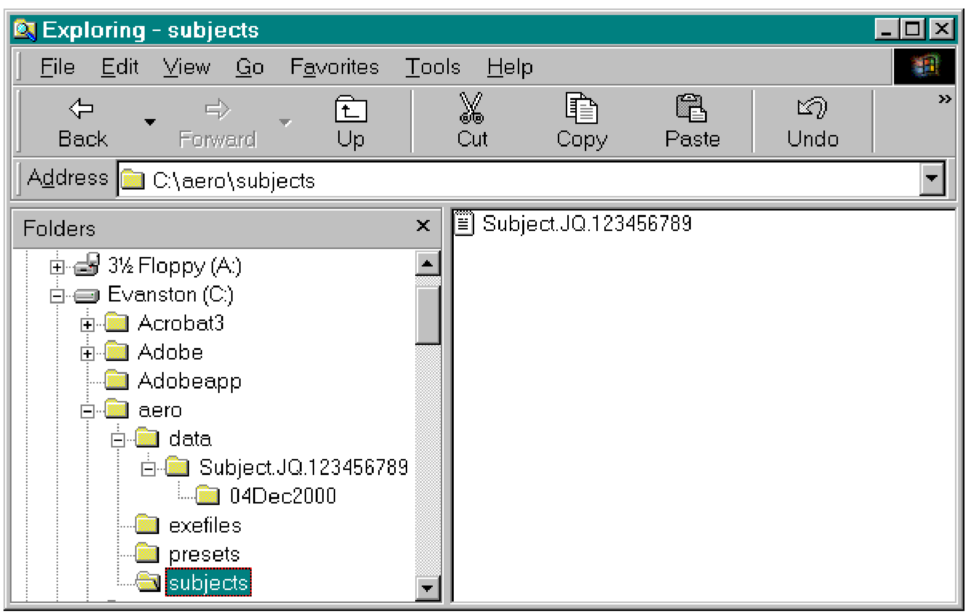


I recommend specifying values for the name fields; last name, first and middle initials, and subject ID are used to generate a directory name on the computer hard drive where all the data for that subject gets logged. Next time you invoke `Files New session` – by activating TF32 or by selecting `New session` from the `Files` menu when TF32 is already running – you can enter the ID `123456789`, press `Locate`, and the subject information will reappear on the form.

You can change the subject name, useful if a subject has a legal name change owing to marriage or other circumstances, although the system will keep the same data-logging directory based on the original name and ID value. The system is keyed on the ID – if you enter a different ID value you are creating a new subject entry while if you change the name and keep the ID, you are working with the same subject.

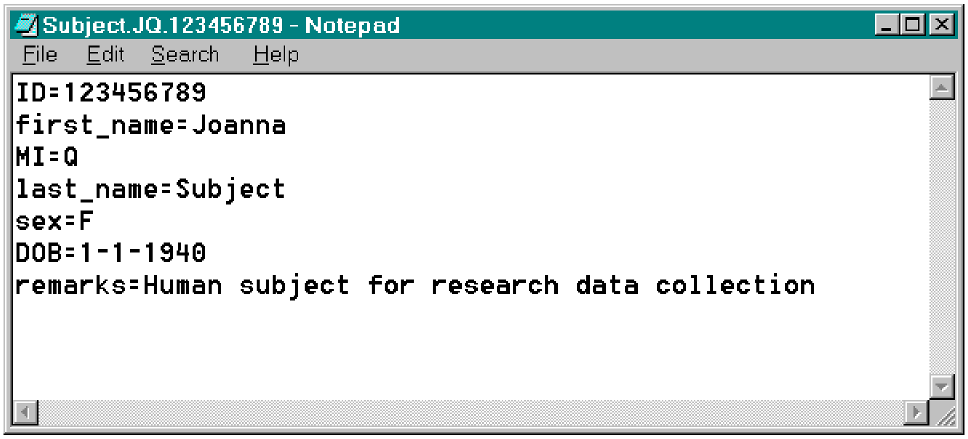
The information in the `Remarks` box is global to all the sessions you create for that subject. The length is limited to what fits in the displayed edit box. Use `Remarks` to enter keywords for a particular subject/patient (belonging to a category, participating in a research trial) to help retrieve that subject based on a keyword. You can change the contents of `Remarks`, but the old contents of `Remarks` are not saved as part of the log entries to an earlier session for that subject. To enter remarks or notes specific to a particular recording session, use the `Annotate session log` command on the `Edit` menu.

Activate Windows Explorer from the Windows `Start` button at this point and click on the nodes `aero`, `data`, and `Subject.JW.123456789` to expand the tree. Now click on the node `subjects` and you will see

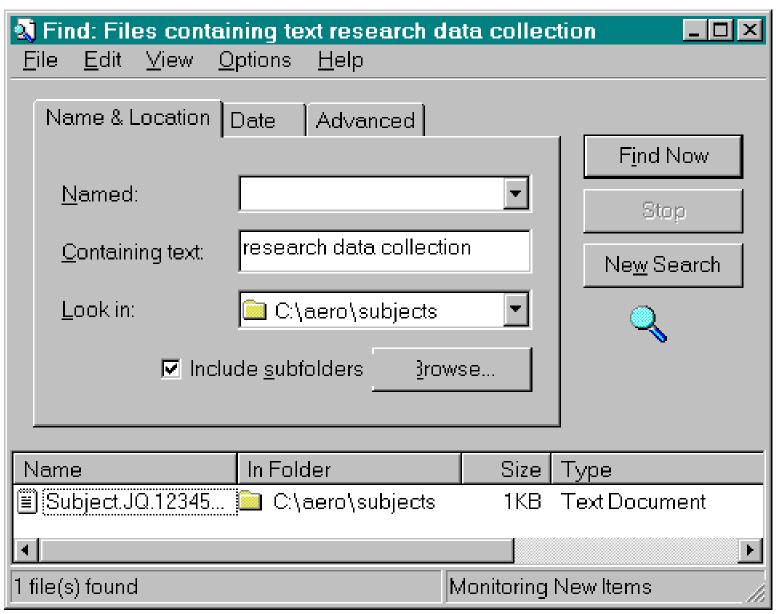


The path `c:\aero\data\Subject.JQ.123456789\04Dec2000` is where all the recordings get saved. The file `c:\aero\subjects\Subject.JQ.12345679.txt` contains the subject description just entered with the `Files New session` dialog.

If you double-click on the file name in the right hand panel of Windows Explorer, you will see



The information contained in this file is usable for retrieving subjects. From Windows Explorer, invoking `Tools Find Files or Folders` and searching for the key phrase `research data collection` in the path `C:\aero\subjects` turned up

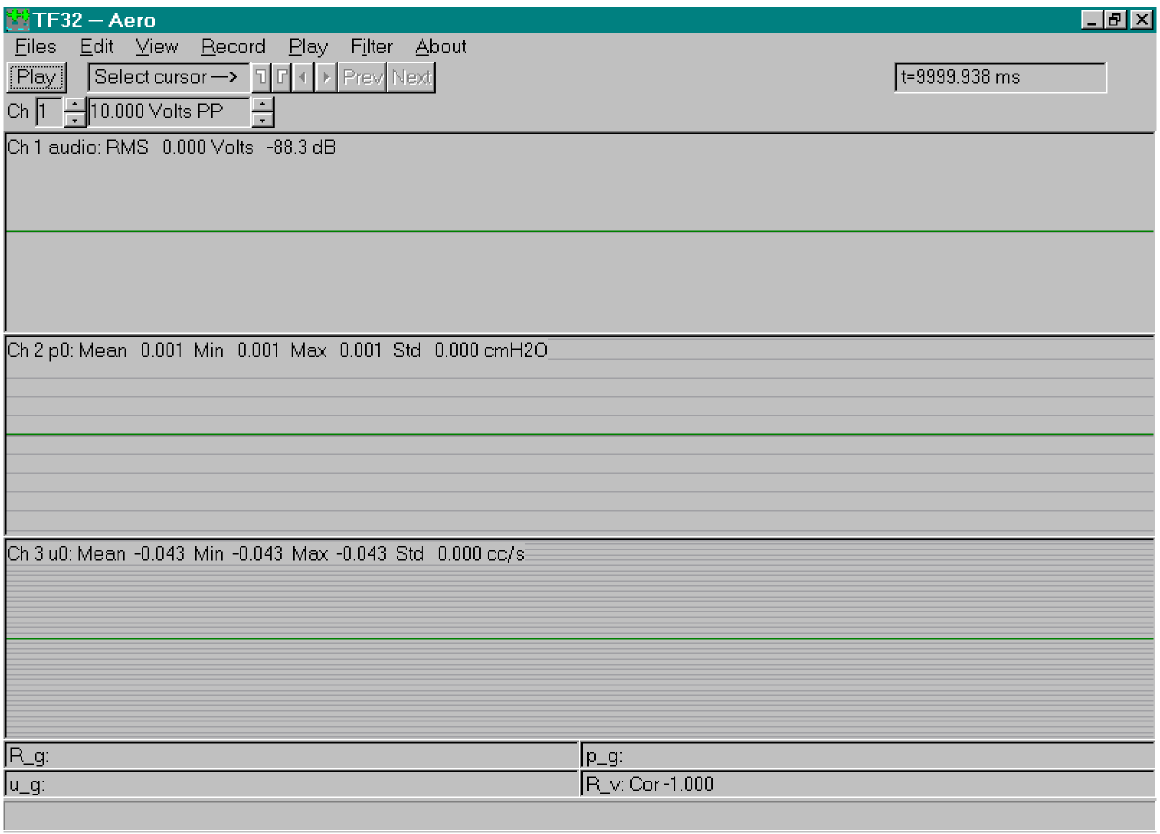


To use this feature, it is important to to decide how you are going to identify each subjects and to select keywords to place in the `Remarks` box. You can change the `Remarks` by invoking `Files New session` from TF32 and reentering a subject's ID.

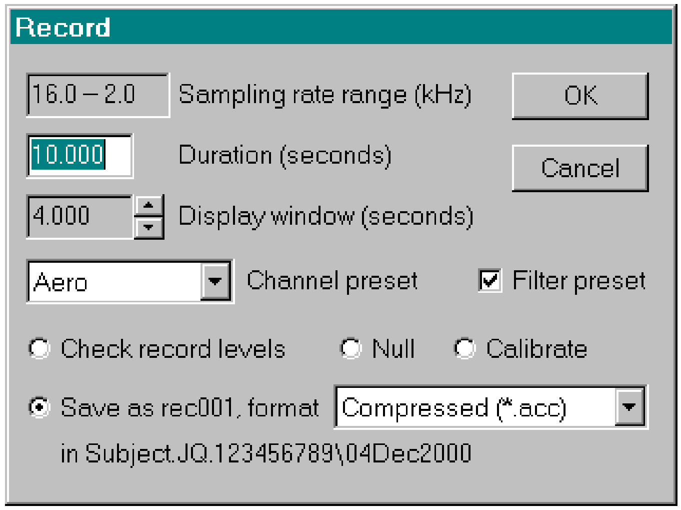
6.1.2 Calibration

It is important to calibrate the offsets and ranges of the pressure and flow channels before logging any data from your subjects. Nulling the channels (sets the offsets) is an action that you should perform as part of each recording session. Calibrating the channels (sets the ranges) is a more involved operation you may need to do "offline" in preparation for a session with a subject.

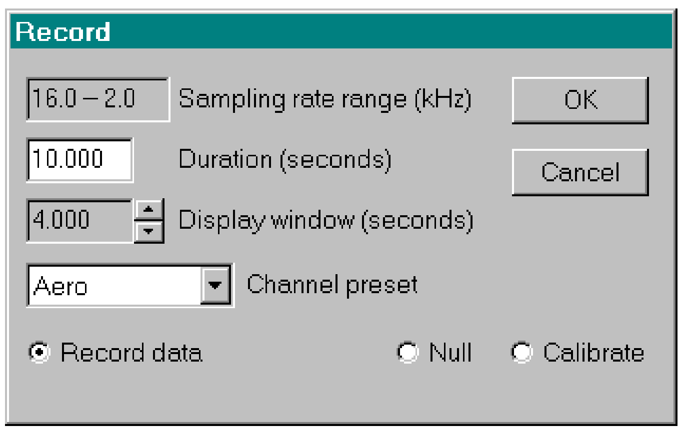
When you activate TF32 by clicking on the `Aero` icon, it asks for an ID. When calibrating, it may be helpful to enter the ID `123456789` to select the "generic subject." If you setting the null as the initial step of conducting a recording session with a subject, enter the ID number for that subject at this time. You will see the set of zero-valued waveform channels



Select `Record` from the main menu and you will see



If instead you see



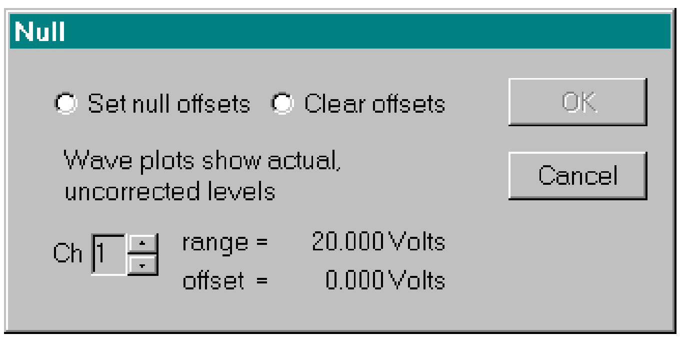
select `Cancel` to close the dialog and select `Files New session` from the main menu.

The first action is to set the null, whether or not you chose to do the full calibration. Make sure the amplifiers are turned on and that the pressure tube and flow mask are receiving ambient pressure and zero flow. Orient the transducers in the same position used to record from a subject: this minimizes the effect of gravity on the transducer readings. Having selected `Record` from the TF32 menu, select the `Null` option in the Record dialog. This action increases the acquisition buffer duration to 200 seconds – enough time to record and let the waveform levels stabilize. Select `OK` or press the keyboard spacebar to begin recording.

The waveforms you see scrolling across the screen show raw, uncorrected waveform levels, allowing you to set the balance by adjusting the offset controls on the amplifier front panel to center each waveform trace about the zero level. It is important to be viewing uncorrected waveform levels when adjusting the amplifier offsets; if the waveform levels already have a software correction for offset, you could be chasing an offset level until your offset is all the way at one extreme of the A/D input range, risking clipping of your recorded signals. Recording with the `Null` option is the only time the display will show uncorrected signal levels – at all other times you should not touch the amplifier offset controls. If you think someone has fiddled with the amplifier offsets, reset the balance following this procedure again.

It is important to make this coarse correction with the amplifier offset controls to maximize the usable signal range of your transducers. If you don't set the balance by adjusting the amplifier offset controls, you risk losing signal to clipping.

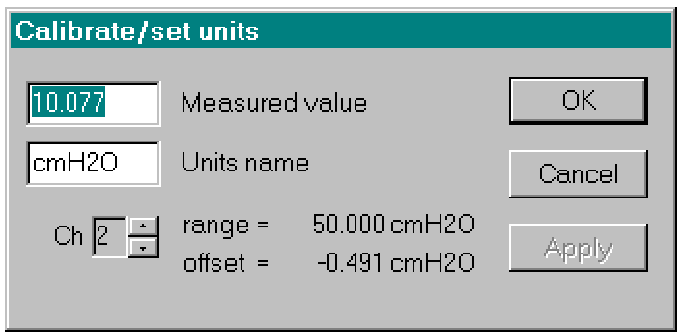
When you are done make small adjustments to the amplifier offset controls to center the waveform levels as best you can, let the acquisition run until the waveform levels stabilize and show the same level across the entire screen. Stop recording by selecting the `Stop` button with the mouse or pressing the space bar on the keyboard. You will see the dialog



The waveforms in the display underneath this dialog should be stable across the entire display window and should be close to the zero-level centers of each plot. If not, select `Cancel` and repeat this procedure, paying attention to the signal balance as you observe the scrolling waveform display during recording. If these conditions are satisfied, select the `Select null offsets` button in the dialog. This action causes the waveform levels in the wave plots to be centered at zero; it also sets the offset values that correct for the true zero reading in each channel that you can observe by working the channel clicker. You may also check to see that each offsets is less than 10 percent of the corresponding range for each channel to guard against clipping problems, but having checked that the waveform levels are not way off from zero before selecting `Select null offsets` will get you close enough. If everything checks out, simply select `OK`, and the balance (hardware correction) and null offset (software correction) are now properly applied to each waveform channel.

To summarize the null procedure, invoke `Record`, select `Null`, check to see that all your transducers are getting zero input, select `OK` to start recording, adjust the offset controls on the amplifier front panel to get a coarse null correction by centering the waveform levels at the zero level, select `Stop`, check `Set null offsets` in the dialog to apply the fine null correction and select `OK`. This procedure sets all channel nulls.

The next action is to calibrate the pressure and flow channels. The best way is to calibrate one channel at a time so you can devote full attention to setting the value of one pressure or flow source at a time. First connect the pressure tube to a source where you can set the pressure to a known value. Select `Record` from the main menu, select the `Calibrate` option in the Record dialog, and select `OK` to start recording. Adjust your pressure source to the desired value, wait for the levels on the scrolling waveform display to stabilize to a constant level across the screen, and select `Stop`. You will see the dialog



In order to read Ch 2, use the channel clicker to select Ch 2 (Ch 1 is audio, Ch 2 is pressure, Ch 3 is flow). The value 10.077 is what the system currently reads for pressure in this example. If the u-tube manometer connected to your pressure source is reading 20, enter the value 20 into the edit box and select `OK`. Now connect the flow mask to the flow source and repeat the process to calibrate flow. Be sure to select Ch 3 when calibrating the flow channel.

To check the offset and range values for each channel, you can invoke `Record` and select `Calibrate` at any time without changing the settings, or you can rerun the null procedure, which is a good idea anyway. These offset and range values are automatically saved in the settings file `aerocal.pre` and are retrieved when you activate TF32 by clicking on the `Aero` icon on the Windows desktop, meaning that calibration settings are remembered between sessions.

Both the Null and Calibrate dialogs display the peak-to-peak (+ or −) range of the calibrated signal for the channel you select with the clicker as well as the offset value used for the software null correction. If the offset is more than 10 percent of the range value, it is best to reset both the balance and null by invoking `Record` and selecting the `Null` option. If the range value is outside of norms, you will have to change the gain settings for the offending channel on the amplifier control panel, and you will need to redo the null and calibrate. Be sure to reset the null after changing the gain and before recalibrating as the gain setting can change the null.

The amplifier gains have been set in the shop for a pressure range of 200 cm H2O (max 100, min -100) and for a flow range of about 4000 cc/s (max 2000, min -2000). You should leave the amplifier gain settings alone unless these ranges are not adequate for the subjects you are measuring. If you keep the amplifier gains at the original settings and if your calibration ranges differ by more than 40 percent from 200 cm H2O and 4000 cc/s, you are 1) getting a leak and hence erroneous readings in your calibration sources, 2) the amplifier gains have been changed, or 3) the transducers or the amplifier system require service. Keep a written record of the amplifier gain settings to check for changes.

Remember to null the channels (sets the correct zero level) before you calibrate (sets the channel range to match a known measurement level). You can set the null later on to correct for transducer drift without having to redo the calibration. Also remember to check `offset` against the `range` values reported in the Null dialog to see if you need to rebalance.

Setting the null is an action you can perform frequently because it only requires you to expose the pressure tube to atmospheric pressure and to remove the flow mask from a source of flow. My recommendation is to

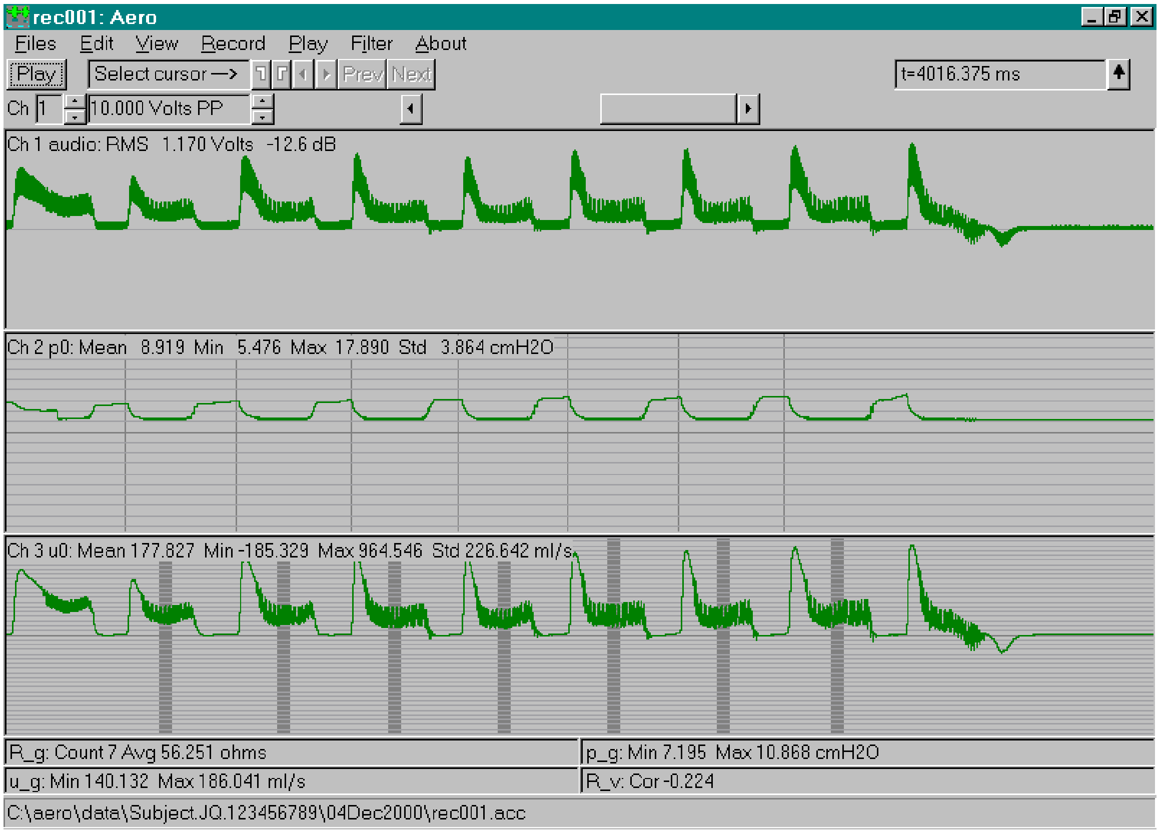
> Null early and null often!

6.1.3 Measuring aerodynamic resistance

Having specified a recording session keyed by subject ID and session date with the `New session` dialog and having calibrated the pressure and flow channnels, you are ready to record waveforms and make measurements. Before proceeding, invoke `Annotate session log` from the `Edit` menu to make a remark about the session you are conducting with that subject or patient (initial evaluation, 6-month follow-up, and so on).

Instructed your subject on the task they are to perform, and with the pressure tube and flow mask in place on the subject, you are ready to acquire waveforms and log acoustic and aerodynamic measurements. Select `Record` from the main menu, verify that the recording duration and length of the scrolling display window are set to usable values, verify that the channel preset is set to `Aero` and the `Save` option has been selected, and select `OK` (or press the space bar on the keyboard) to begin recording. Terminate recording by pressing the space bar or by selecting the `Stop` button in the upper left hand corner of the TF32 main window with the mouse. It is often useful to stop recording when the subject has produced the task correctly and the desired waveform traces are seen in the scrolling display window.

Recording a repeated CV task (for the measurement of laryngeal resistance) produces a display that looks like



Ch 1 contains the waveform plot of the audio signal. The numeric readout in the upper left of the waveform plot gives the RMS and dB level of the audio signal for the entire 4 second display window. Ch 2 contains the pressure trace. The trace shows the tail end of one pressure pulse followed by 8 complete pressure pulses. The numeric readout for this channel gives mean, min, max, and standard deviation of pressure for the entire 4 second display window. Ch 3 contains the flow trace. The pressure pulse for each CV is followed by the flow peak prior to the onset of voicing followed by the sustained flow for the voiced interval.

The bottom panel contains numeric readouts labelled `R_g` (automatic calculation of laryngeal resistance for the CV task), `p_g` (min and max values of the pressure pulse peak-to-trough values), `u_g` (min and max values of the average flow in the center of the voiced interval minus flow at the pressure peak) and `R_v` (automatic calculation of velar port resistance from intraoral pressure, nose cone flow recording). The `R_g` calculation requires a complete CVC cycle in order to identify the middle of the voiced interval. The automatic algorithm has identified 7 complete CVC cycles.

The automatic algorithm for laryngeal resistance measures flow by taking a Hamming-window weighted average over a 40 ms interval between pairs of pressure pulses. The interval between pressure pulses is allowed to have a minimum of 40 ms, a maximum of 1000 ms – intervals outside those ranges are rejected as being valid CV's. For intervals between pressure pulses up to 240 ms, the flow-measurement interval is centered between pressure pulses. For longer intervals between pressure pulses, the VOT (time from end of pressure pulse to start of flow measurement interval) is limited to 100 ms.

To help determine that the CV tracker is working correctly, vertical lines mark the positions of the pressure pulses used to calculate laryngeal resistance while the dark bands mark intervals during sustained voicing used to compute averages of voiced flow. The last CV is not marked because the algorithm requires complete CVC cycles. If the marked pressure peaks and voiced flow intervals are not where they should be, scroll the display or move the waveform cursors to select an interval where the algorithm is able to make a correct identification or reinstruct your subject on the task and record another trial.

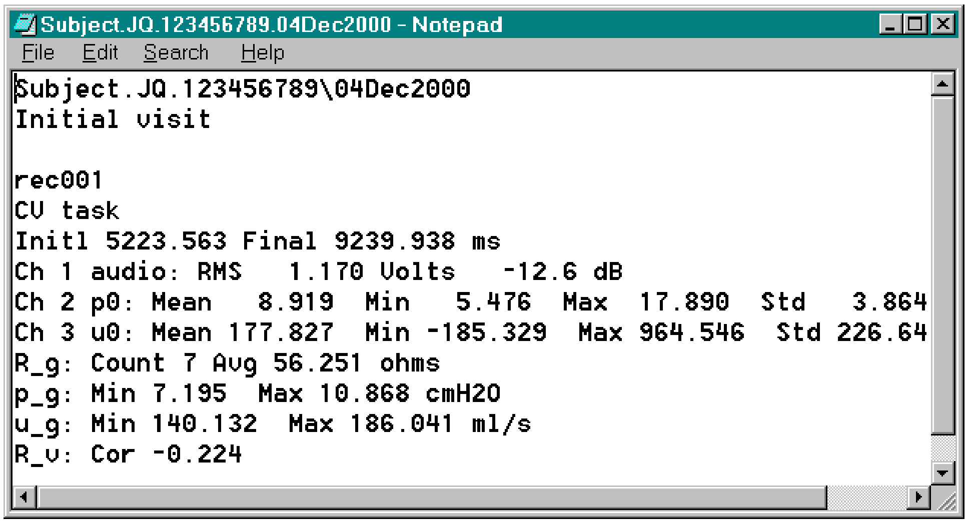
Another "sanity" check on the automatic algorithm is the relation between the `p_g` and `u_g` ranges and the calculated resistance. For pressure `p_g` in cm H2O, flow `u_g` in cc/s (or ml/s), resistance is reported in acoustic ohms, which is pressure in dynes per square cm divided by flow in cc/s. The conversion from cm H2O to dynes per square cm requires multiplying by 981 cm/second squared, the acceleration of gravity. The ohms value should be about 981 times the pressure in the center of the range divided by the flow in the center of the range.

The `R_v` calculation expects flow to correlate with pressure – the readout reports a correlation coefficient of 0.435 (below the threshold of .7), so it does not report a velar resistance value for the laryngeal CV task (oral pressure, oral flow). This calculation will find a correlation above threshold and report a resistance for the velar CV task (oral pressure, nasal flow, subject exhibiting velar leakage).

If the display window does not capture the desired production, you can drag on the scroll bar above the waveform plots to center the display on that production. Or you can click in the up arrow control in the upper right hand corner: click once to zoom out to the entire waveform buffer duration, click a second time to remove the cursors marking the interval you zoomed out of. Drag the left mouse button over the display to place the left cursor, drag the right mouse button over the display to place the right cursor to select a waveform interval. Press the down arrow that appears in the upper right corner of the main window to zoom in to that interval.

Once you have selected the desired waveform interval, all the numeric readouts will automatically update to measure that interval. Invoke `Edit Annotate session log` (select `Annotate session log` from the `Edit` menu) to annotate or make a remark in the session log file about the waveform record you have just collected. Select `Edit Log readouts` to log the numeric readouts, or select `Edit Update and view log` to log the numeric readouts and view the contents of the log file with Notepad. You must exit Notepad at this point to resume collecting more records with TF32. If you do not explicitly log a measurement, the measurements that appear on the screen will be logged automatically when you collect the next record with `Record` or exit TF32. This insures that at least one measurement is logged for every trial.

If you had selected `Update and view log` from the `Edit` menu (does the same thing as first selecting `Log readouts` followed by `View log`), you will see




```
Subject.JQ.123456789\04Dec2000
Initial visit

rec001
CV task
Initl 5223.563 Final 9239.938 ms
Ch 1 audio: RMS   1.170 Volts   -12.6 dB
Ch 2 p0: Mean   8.919  Min   5.476  Max  17.890  Std   3.864
Ch 3 u0: Mean 177.827  Min -185.329  Max 964.546  Std 226.64
R_g: Count 7 Avg 56.251 ohms
p_g: Min 7.195  Max 10.868 cmH2O
u_g: Min 140.132  Max 186.041 ml/s
R_v: Cor -0.224
```

The log contains the initial annotation "initial visit," it also contains the annotation "CV task" made immediately after collecting the first record. The measurements were entered in the log by selecting `Edit Update and view log` in order to view the log file. You can copy this file to the printer by selecting the print command in Notepad. Remember to exit Notepad to resume TF32.

The TF32 program offers considerable flexibility in making measurements and obtaining hard copy of displays. Select `View Wave plots` from the main menu to change the numeric readout (options are endpoint values, stats, and RMS dB) or to change the y-axis grid lines for any of the waveform plots. Select `View Open` to add analysis displays for pitch, audio waveform dB contour, time-frequency spectrogram.

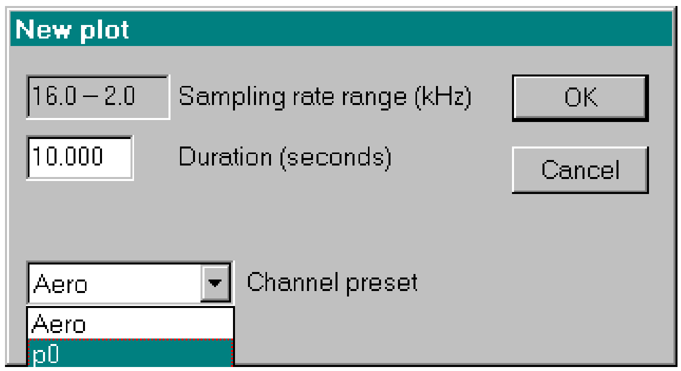
Among the options on the `View Open` submenu are `R auto` and `R cur`. The `R auto` option for automatic calculation of resistances is the default – the measurements will revert to this option every time you collect a record. The `R cur` option allows you to measure resistance by placing the waveform cursors. For laryngeal resistance, place the left cursor on the pressure peak, place the right cursor about 20 ms after to phonation onset. For velar resistance, place the cursors at locations with differing amounts of flow and pressure. The manual measurement of resistance gives a way of getting readings on data where the automatic algorithms fail to track.

There are two ways to transfer images of the TF32 displays into a word processor document. To get the entire image with menus and window borders, press the `Alt` and `Print Scrn` keys at the same time – then paste into your document. The images in this manual were transfered with this technique – you may find this useful for producing your own training manuals for your particular facility. The transfer detailed waveform or spectrogram plots without the menus and window borders, select `Edit Copy image to clipboard` from the TF32 main menu.

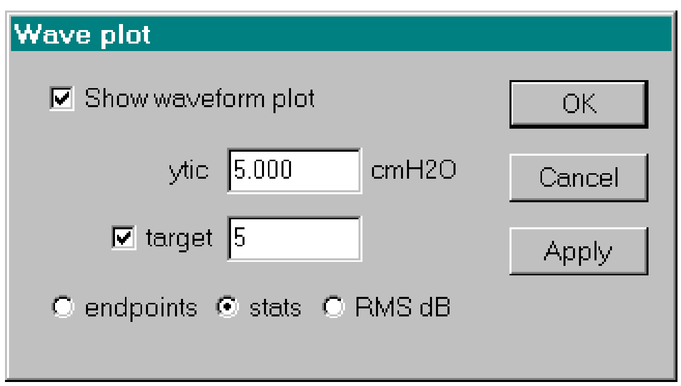
6.1.4 Pressure-matching task

The pressure-matching task requires 1) configuring the display for a single pressure channel, 2) setting a pressure target level in the waveform display, 3) recording the subjects response to the target level, and 4) placing cursors and logging measurements of the subject's responses.

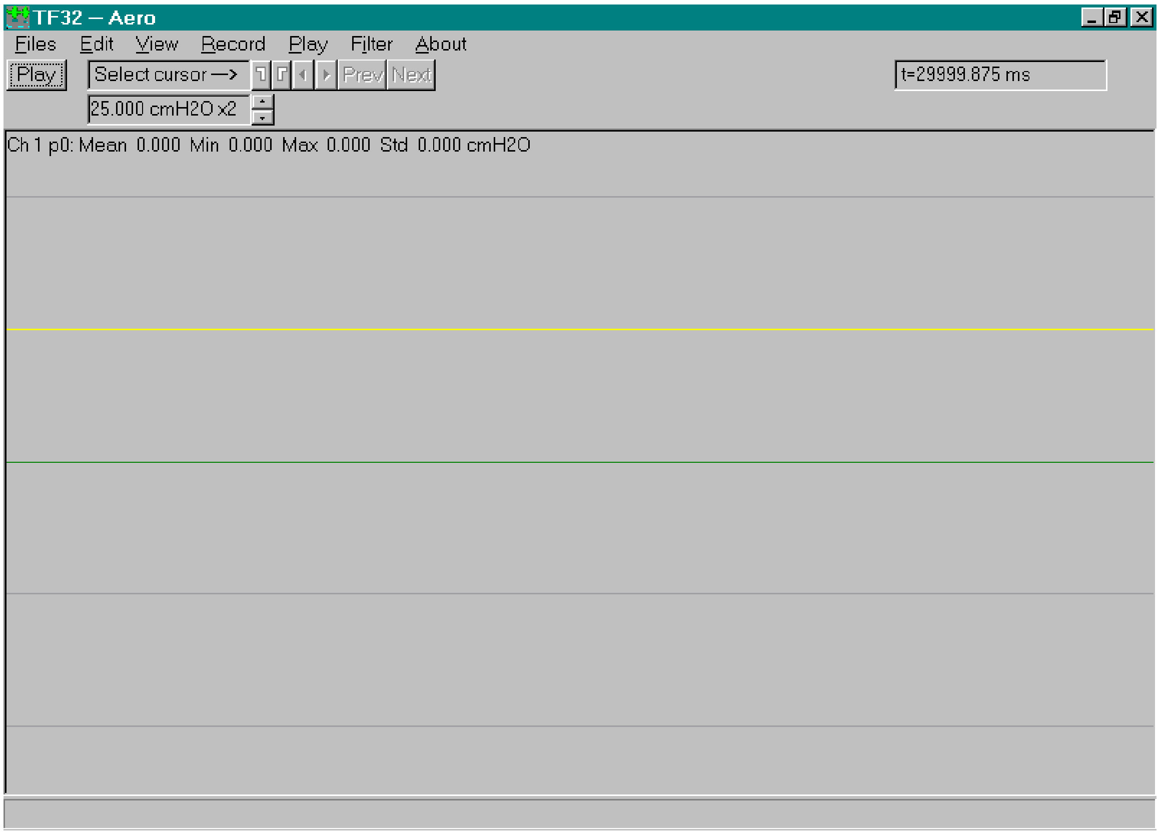
Configure the display for pressure matching by selecting `Files New plot` from the TF32 menu and selecting the `p0` option from the `Channel preset` list box according to



Next, select `View Wave plot` from the main menu, check the `target` box, and enter a value for the target level according to



You will produce a display looking like



Adjust the displayed range of the display by working the clicker next to

```
25.000 cmH20 x2
```

The `x2` means that the display is scaled up by a factor of 2.

Select `Record`, set the record duration to the desired value, and select `OK`. When recording is complete, select the time intervals you want to measure with the waveform cursors and log measurements by selecting from the `Edit` menu.

To revert to audio-pressure-flow measurement, select `Files New plot`, select `Aero` from the `Channel select` list box, and select `View Open R auto` to bring back the resistance readouts.

7 Installing TF32

7.1 Hardware requirements

TF32, Time-frequency analysis for 32-bit Windows, requires a computer using Windows 95, 98, NT, or other compatible version or emulation of Windows. TF32 will not run under Windows 3.1 or earlier versions.

A recommended minimum system has 64 MB RAM, 500 MHz Celeron processor, mid-range performance PCI graphics card, and 17 inch monitor operating at 1024 by 800 resolution and at least an 80 Hz refresh rate. That amount of RAM permits recording 44.1 kHz stereo for up to 4 minutes with smooth scrolling of the display; increased RAM, sampling a single channel, or sampling at a lower rate all give a proportional increase in recording time. A 500 MHz Celeron permits smooth scrolling of dual spectrogram displays while recording 44.1 kHz stereo with a 1 second scrolling window. A faster processor will permit smooth scrolling with real-time display of pitch or the time-slice spectrogram.

TF32 is available with three levels of support for A/D and D/A hardware.

Demo level supports waveform playback only, monaural or stereo, through any Windows-compatible sound card.

Basic level supports both recording and playback of monaural or stereo audio through any Window-compatible sound card.

Lab Automation level supports recording up to 16 DC-coupled channels at differential sampling rates (audio high rate, physiology channels lower rate) from the Data Translation family of A/D cards through the DT Open Layers software interface supplied by Data Translation. Also supports recording and playback through the Windows-compatible sound card. All waveform playback is done through the sound card – does not require the D/A functions on the Data Translation card.

TF32 supports sound cards compatible with the Windows Multimedia interface. These cards permit recording and playback of up to two AC-coupled channels. The low-pass anti-alias filters are often built into these cards, or one is sampling at a high enough rate (22 kHz or higher) where speech is naturally bandlimited and aliasing is not a concern. These cards vary in their audio specs – signal-to-noise, audio low-frequency and high-frequency cutoffs, the use of anti-alias filters, allowed sampling rates – so please check with the manufacturer regarding specifications.

Some applications require DC-coupled channels, sampling more than two channels, or sampling at differential rates. Acquisition of an acoustic signal at a high rate, DC-coupled flow and pressure signals at lower rates, is such an application. Conventional AC-coupled audio-level sound cards are not suitable for these applications. The

Laboratory Automation level of TF32 supports Data Translation A/D cards meeting these requirements.

TF32 supports the use of a Data Translation card by automatically linking to a driver called DT Open Layers. You need to obtain the Open Layers driver from Data Translation and install it on your computer. DT Open Layers is supplied by Data Translation with the purchase of one of their cards or may be downloaded from the Web site `www.datx.com`. The dynamic link library (DLL) files `oldaapi32.dll` and `olmem32.dll` are installed on your computer when you install Open Layers for your particular Data Translation card. TF32 automatically links to these DLL files whenever it runs. If these files are absent, TF32 will still operate and record and play from your system sound card, but TF32 will not operate your Data Translation card.

In principal, TF32 should work with any Data Translation card for which DT Open Layers has been installed provided the card supports "continuous performance" (either with or without the use of the DMA feature), digitizes at either 12 or 16 bit resolution, and has sufficient aggregate sampling rate (sum of rates of all sampled channels) for your application. The card does not need to support playback because all TF32 playback will take place through the sound card. So far, TF32 has been tested with the DT2821, an older-model 12 bit card for which you will have to get drivers from the Data Translation Web site, the DT9803, a new 16-bit card that works through the USB port on either a desktop or portable computer, and the DT301, a new 12-bit card that uses a PCI slot inside your computer.

The DT9803 connects to the USB port on the outside of your computer, making it usable with either a desktop or laptop computer – all other cards require a desktop computer with internal card slots. The Data Translation USB driver is restricted to Windows 98 (95 and NT are not compatible) – check with Data Translation for any changes on this.

If you use TF32 with a sound card, it is usable with any of the microphones, speakers, and line-level inputs and outputs that are compatible with that sound card. Typically the anti-aliasing filtering is either built into the sound card, or you are sampling at rates (22.05 kHz or higher) where aliasing is no longer audible. If you are recording audio with a Data Translation A/D card, you will need to the acquire transducers, preamplifiers, and anti-alias filters needed for your application.

7.2 Installing TF32 under Windows

The TF32 software program has a single executable program file – `tf32.exe`. TF32 does not require DLL driver files, although TF32 Laboratory Automation will automatically link to the DT Open Layers DLL files installed with the software for your Data Translation card. The Data Translation DLL files are *not* included with TF32.

TF32 may be activated from the Windows DOS prompt or from the `Run` command selected after pressing the Windows `Start` button. Creating a desktop icon allows starting TF32 with a mouse click. Right-button mouse click on the Windows desktop, select `New` followed by `Shortcut`, and specify the path to wherever you decide to keep the file `tf32.exe`. Right click the TF32 icon with the mouse and select `Properties` to select a different startup directory from where you keep `tf32.exe` if you desire.

Any file you can open in TF32 (the menu command `Files Open`) can be opened automatically when TF32 starts – simply list one or more of these files on the TF32 command line in the order to be opened. Right click the TF32 icon and select `Properties` to add these names to the TF32 command line. You must include the file extension when listing it on the command line.

Opening a waveform file displays waveform plots and automatically updates the analysis displays (pitch, spectrogram) computed from those waveform plots. Opening a waveform list file (`.lst` extension) plots the first waveform segment in the list and enables use of the `Prev` and `Next` buttons for indexing through the list. Opening a preset file (requires Lab Automation level) customizes the waveform plots, analysis displays, and specifies A/D channels and sampling rates for the `Record` command.

To open one or more files on startup, list the file names on the TF32 command line, from left to right in the order to be opened.

7.2.1 Installing TF32 Lab Automation from the CD-ROM

The TF32 Lab Automation CD-ROM has an `install` command for copying the `tf32.exe` program file and the necessary preset (`.pre` extension) files for configuring TF32 to perform voice aerodynamic measurements. This installation procedure will also create a Windows desktop icon called `Aero` which you may click to activate TF32 with this configuration.

Install the Data Translation driver for your A/D card first. Next, place the TF32 Lab Automation CD-ROM in your CD-ROM drive, change to the root directory of your CD-ROM drive (typically `d:\`) and invoke the `install` command. This procedure creates or updates a directory called `aero` on your `c:` drive, copies files, and creates the `Aero` icon.

7.3 Preset file

The preset file is an ASCII file that customizes the waveform plots, analysis displays, and A/D channels of TF32. This feature of the Lab Automation level allows TF32 to acquire multiple channels at differential sampling rates according to one or more preprogrammed configurations and to provide one or more configurations of analyses to go with each channel configuration. The preset file is required to specify A/D channel configurations when recording from a Data Translation A/D card; the preset file is optional when recording from the Windows multimedia card.

The TF32 Lab Automation CD-ROM includes the following preset files to configure TF32 for voice aerodynamic measurement

`aero.pre` is the preset used when activating TF32 with the `Aero` icon.

`aeroview.pre` is a preset that may be used with TF32 when viewing already-recorded data – this preset locks out the `Record` command.

`aerocal.pre` specifies calibration information, range and offset, for the A/D channels. This file is automatically updated by TF32 when you change the signal offset with the `Record` command (`Null` option on the `Record` dialog) or change the signal range (`Cal` option on the `Record` dialog).

Installing TF32 from the Lab Automation CD-ROM copies these files to the directory `c:\aero\presets`.

A preset file may be loaded by opening it with the `Files Open` command or by listing it as a file name parameter on the command line that activates TF32. Installing TF32 from the Lab Automation CD-ROM creates an icon called `Aero` that loads the Aero preset from the command line – that preset is the default when TF32 is activated by clicking on the Aero icon.

The first line of the preset file contains the keyword

```
TF32PresetList
```

identifying the contents of the file as configuration information instead of waveform data. That way, opening a waveform file with `Files Open` will load that waveform into the TF32 display while opening a preset file with `Files Open` will load that preset, making the preset channel configurations selectable from the `Files New plot` or `Record` dialogs and making the preset analysis configurations selectable from the `View Open` submenu.

Subsequent lines in the preset file specify keyword entries, which appear on a single lines, or block definitions, which appear on multiple lines. Each block consists of header on a single line, keyword lines, nested blocks, followed by a single line that closes the block. Blank lines along with spaces preceding or following a keyword entry or block header are ignored; spaces within a keyword entry or block header are significant.

The entries in the preset file specify the *properties* of software *components* that make up the TF32 display. The entries are not statements in a programming language like C or Pascal – there are no `if`, looping statements, or subroutines. Writing a preset file is not programming in the sense of a general purpose language like C or even a scripting language like Java script. Rather, it is "programming by configuration": specifying selectable lists of channel (plot preset) and analysis (view preset) configurations.

The preset file is the means by which TF32 Lab Automation is customized for specific data acquisition requirements. A text file instead of a graphical interface is used to provide this customization because a data acquisition configuration can be complex, especially in the case where multiple channels are acquired at differential rates.

The information provided is a guide for understanding the contents of the preset files along with making modifications to those preset files. When customizing TF32 Lab Automation, it is best to use the preset files on the CD-ROM as a starting point, to make incremental change to those files, and to test each change.

The following page shows a truncated version of the preset file `aero.pre` that illustrates the use of keywords and nested blocks. This particular example defines a selection named `Aero`, appearing in the `channel preset` drop list of the `Files New plot` and `Record` dialogs, which specifies a 3-channel waveform plot – audio, p0 (pressure trace), and u0 (flow trace) – for recording or display. Scale, tic marks, and numeric readout mode are specified for the pressure and flow traces, and automatic measurement of glottal resistance is applied to those traces. A selection named `1. R auto` appears on the `View Open` submenu: if the glottal resistance analysis window is closed by the user, it may be reopened by selecting from the `View Open` menu.

```
TF32PresetList
ShowWindowMaximized
NewSession
open_fname=c:\aero\presets\aerocal.pre

preset:plot=Aero
    default_view=2
    ADname=DT301(00)
    NCHAN=3
    NSCAN=16
    sample_Hz=16000.0
    CH=1
        OCHAN=1
        DCHAN=2
        CHname=audio
    CH=2
        OCHAN=2
        DCHAN=16
        CHname=p0
    CH=3
        OCHAN=4
        DCHAN=16
        CHname=u0
end_preset

preset:view
    wave_plot:CHname=p0
        readout=stats
        ytic=5.0
        scale=11
    end_wave_plot
    wave_plot:CHname=u0
        readout=stats
        ytic=50.0
        scale=11
    end_wave_plot
end_preset

preset:view
    preset_readouts
    view_name=&1. R auto
end_preset
```

```
preset:view=&1. R auto
    ReadoutWnd:CV_resistance
        select_p
            CHname=p0
        select_u
            CHname=u0
    end_ReadoutWnd
end_preset
```

The keyword `ShowWindowMaximized` maximizes the TF32 main window and the keyword `NewSession` invoke the `Files NewSession` menu command to allow entry of subject information upon the activation of TF32 from the Aero icon. The keyword `open_fname=` specifies a file to open after opening the current preset file. It could be a waveform, but in this example it appends the preset file `aerocal.pre` containing the A/D channel calibration settings.

The sections bracketed by `preset:` and `end_preset` are preset definition blocks. Each definition block contains keywords or nested blocks specific to the kind of definition.

The block starting with `preset:plot=Aero` defines a plot preset. A plot preset is a configuration of channels in the display selectable from a drop list from either the `Files New` or the `Record` commands. `Aero` is the name of that plot preset that appears in the drop list for selecting that configuration of channels. The first plot preset specifies the initial configuration of the waveform channels when a preset file is opened.

The blocks starting with `preset:view` defines an unnamed view preset. A view preset is the configuration of analysis that is applied to the waveform channels selected with a plot preset. The block starting with `preset:plot=&1. R auto` is a named view preset. Named view presets appear as entries appended to the `View Open` submenu while unnamed view presets are hidden from that menu but may be selected by other means.

The first view preset, named or unnamed, specifies the initial analysis configuration when a preset file is opened. The second view preset (an unnamed view), is specified by the `default_view=2` keyword in the `Aero` plot preset. The typical pattern is to use a first unnamed view preset to specify a global configuration – the `wave_plot:CHname=` nested blocks specify readout, tic mark, and scale settings for named channels – followed by an ordered set of unnamed view presets to be selected by the `default_view=` keyword for those plot presets that require a default view preset. The named presets that appear in the `View Open` submenu then follow.

A view preset may refer to other view presets. The second (unnamed) view contains the keyword `preset_readouts`, which selects the channel readout, tic mark, and scale settings designated by the first, default view preset. The second view also contains the entry `view_name=&1. R auto`, selecting the named view preset with the name `1. R auto`.

Unlike with block-structure programming languages such as C or Pascal, semicolons are not permitted; each entry in the definition block must go on a separate line as shown in the example. You may precede each entry with blank spaces, and you may use blank lines, as used in the example, to format the text for readability. Spaces within a definition entry, however, are significant; you must not introduce spaces around the `=` or `:` formatting characters in

```
preset:plot=&1. R auto
```

and the spaces within the text label

```
&1. R auto
```

are part of that label.

7.3.1 Contents of the preset file

The preset file may contain the keywords

`ShowWindowMaximized` maximized the TF32 display window,

`NewSession` invokes the `Files New session` menu command on startup or on opening the preset file to permitting entering subject data for a recording session,

`ViewMode` puts TF32 in view mode, which allows analysis but disables recording,

`open_fname=`*fname* invokes the `Files Open` command on *fname* – used for chaining to another preset file containing the `cal` preset definition, but also may be used for opening an initial waveform.

The preset file may contain preset definition blocks. These blocks start with one of the following and end with `end_preset`:

`preset:plot=`*plotname* specifies how channels get mapped from the A/D to the waveform plots. Information about sampling multiple channels at differential rates is included here. Multiple `plot` presets are permitted. The first `plot` preset specifies the initial waveform plots on startup. The `plot` presets are selectable with the `Files New plot` dialog, which places zero-valued wave plots on the screen. This allows selecting analysis displays from the `View` waveform to be computed from those waveform plots in advance of recording. The display is now configured for viewing scrolling displays of acquired waveforms along with real-time updates of the analysis traces when you invoke `Record`. The `plot` presets are also selectable from the `Record` dialog box.

`preset:cal` defines a list of channel names (`CHname=` entries) along with calibration values, range, offset, and units name, for those channel names. Only one `cal` preset is allowed. The `CHname=` entries match A/D channels in the `plot` preset with calibration settings in the `cal` preset. This allows multiple `plot` presets to share the same calibration settings for the channels they share in common.

The Aero configuration included with TF32 Lab Automation places the `cal` preset in the separate file `aerocal.pre` and links to it from the preset file `aero.pre` with a keyword entry `open_fname=`

`preset:view` defines an unnamed view preset specifying analysis displays and channel settings selectable from the `View` menu. Multiple `view` presets are allowed, and the first `view` preset is the initial configuration on startup or upon opening a preset file. While the `plot` preset specifies how which waveforms get acquired, the `view` preset specifies how waveforms are displayed and analysed. A `plot` preset can select a view preset in the order it appears with the `default_view=`*n* keyword where *n* is a number starting at 1.

`preset:view=`*viewname* defines a named view preset where *viewname* denotes a menu label appended to the `View Open` submenu.

`preset:filt` specifies one or more filters required to condition waveforms input from the A/D before taking measurements from those waveforms. While only one `filt` preset definition is allowed, the `filt` preset may contain one or more `filter` definitions keyed to channel names (`CHname=` entries). Such a filter matches up with the A/D channel specified in a `plot` preset with the same `CHname=` entry.

7.3.2 Plot preset

The file `aero.pre` contains multiple plot preset definitions, the first of which is Aero

```
preset:plot=Aero
    default_view=2
    ADname=DT301(00)
    prompt=c:\aero\presets\prompt.wav
    save_format=cmprs
    NCHAN=3
    NSCAN=16
    bufsec=10
    dspsec=4
    sample_Hz=16000.0
    CH=1
        OCHAN=1
        DCHAN=2
        CHname=audio
    CH=2
        OCHAN=2
        DCHAN=16
        CHname=p0
    CH=3
        OCHAN=4
        DCHAN=16
        CHname=u0
end_preset
```

The other plot presets follow the same pattern.

This preset, which specifies a 3-channel plot containing the `audio`, `p0` (pressure trace), and `u0` (flow trace) channels, is listed as the `Aero` entry in the `channel preset` list box of both the `Files New plot` and `Record dialogs`. Since this preset is the first plot preset, this 3-channel plot appears when you invoke TF32 by clicking on the Aero icon or if you explicitly load `aero.pre` with the `Files Open` menu command.

This preset also specifies the second view preset definition in the file as the default set of analyses that go with this plot. The specified A/D is the Data Translation DT301, and if you have only one Data Translation A/D in your computer, it is named `DT301(00)`. It is important to edit the `ADname=` entry if you are using a different Data Translation A/D model or if the Data Translation Open Layers installation has assigned the A/D a different numeric index.

The prompt waveform `c:\aero\presets\prompt.wav` will be played prior to collecting each waveform record in automatic logging mode, selected with the `Files New session` command. You can record your own audio prompt to substitute for `prompt.wav`.

The save format for automatic logging is the compressed format. Choices for `save_format=` are `default`, `cmprs`, `SPHERE`, `RIFF`, or `nohdr`. The `default` or `cmprs` formats are required in order to save calibration information into archived waveforms. The `SPHERE` (used in the TI/NIST/MIT "TIMIT" database), `RIFF` (the Microsoft WAV format), and `nohdr` (headerless binary) formats are provided for exporting waveforms to other programs, but their headers do not provide for calibration settings, and saving into those formats will lose any signal calibration and offset correction.

The `bufsec=10` and `dspsec=4` specify that the default recording buffer is 10 seconds long and the default scrolling window into that buffer displays 4 seconds. You can override these defaults from the `Record` dialog. The remainder of the parameter setting pertain to differential-rate sampling.

During acquisition, 3 channels are sampled, and the channels with names `audio`, `p0`, `u0` are recorded into Channels 1, 2, and 3 of this display. The mapping between named channels and hardware A/D channels is specified by the cal preset.

The channels are recorded at differential rates, the highest of which is 16 kHz, the rate assigned to `CH=1` or `audio`. The keyword `NSCAN=16` specifies that there are 16 slots in a "round-robin" channel scan, and how a specific channel is assigned slots in that channel scan determines its sample rate relative to the highest rate channel. This number must be within the scan limit for the particular A/D card you are using – some models allow longer scans than others.

`CH=1`, with `OCHAN=1` and `DCHAN=2`, is loaded into the channel scan slots starting with slot 1 and into every 2 slots:

```
1 _ 1 _ 1 _ 1 _ 1 _ 1 _ 1 _ 1 _
```

where `_` denotes a slot left free for other channels.

`CH=2`, with `OCHAN=2` and `DCHAN=16`, is loaded into the channel scan slots starting with slot 2 and into every 16 slots – it is only loaded once into the 16 slot scan:

```
_ 2 _ _ _ _ _ _ _ _ _ _ _ _ _ _
```

Since it appears 8-times less often than Channel 1, the sampling rate for Channel 2 (p0) is $16/8 = 2$ kHz.

`CH=3`, with `OCHAN=4` and `DCHAN=16`, is loaded into the channel scan slots starting with slot 4 and into every 16 slots – it is only loaded once into the 16 slot scan:

```
_ _ _ 3 _ _ _ _ _ _ _ _ _ _ _ _
```

The sampling rate for Channel 2 (p0) is also $16/8 = 2$ kHz.

The final channel scan with all three channels loaded looks like

```
1 2 1 3 1 _ 1 _ 1 _ 1 _ 1 _ 1 _
```

Viewing this as the childhood game "Duck-duck-goose", each channel is "called on" at equal intervals to guarantee a uniform sampling rate, and Channel 1 (the 16 kHz rate channel) is called on 8 times more frequently than Channels 2 and 3, which only get called on once each scan. The dashes denote empty slots where no one is called upon – these slots are free for additional channels if needed.

7.3.3 Cal preset

The cal preset is contained in the separate file `aerocal.pre` and is linked from the main preset file `aero.pre` with the keyword command

```
open_fname=c:\aero\presets\aerocal.pre
```

The file `aerocal.pre` contains the entries

```
TF32PresetList

preset:cal
    CHname=p0
        A/D_CHAN=0
        RES=12
        range=200.0
        offset=0
        units_name=cmH2O
    CHname=u0
        A/D_CHAN=1
        RES=12
        range=4000
        offset=0
        units_name=ml/s
    CHname=audio
        A/D_CHAN=2
        RES=12
        range=20.000000
        offset=0
        units_name=Volts
    CHname=u1
        A/D_CHAN=3
        RES=12
        range=4000
        offset=0
        units_name=ml/s
end_preset
```

This cal file containing the above cal preset species four channels: p0 for pressure trace, u0 for narrowband flow trace, audio for the wideband microphone signal, and u1 for wideband flow for use in glottal inverse filtering, and it associates the A/D channel (starting at 0), the significant of bits of sample resolution, the range and offset values used in calibration and null correction, and the units name for that channel.

The idea is to maintain a fixed mapping between A/D channels and hardware devices: `A/D_CHAN=2` is always the audio channel, regardless of whether a particular plot preset selects only the audio channel or selects the audio channel in combination with other channels.

The `RES=`*res* keyword specifies the number of significant bits for a given hardware channel. If that hardware channel is connected to a 12 bit A/D, `RES=12` should be specified. For a 16-bit A/D, `RES=15` is the maximum that should be specified as TF32 only retains the most significant 15 bits to allow 1 bit of headroom for filtering signals. The RES parameter is important when saving waveforms in the lossless data-compressed format – a lower RES will result in higher levels of compression.

The range and offset values will be changed and the file `aerocal.pre` will be rewritten whenever the Cal or Null is set from the `Record` dialog. That calibration and null correction will be saved on disk between activations of TF32 until those settings are changed by rerunning Cal or Nul.

7.3.4 Elements of the view preset

An unnamed view preset definition block is delimited by

```
preset:view
end_preset
```

while a named view preset definition block is delimited by

preset:view=*viewname*
end_preset

The first view preset, unnamed or named, is selected when the preset file is opened while subsequent view presets may be selected from a plot preset by including the `default_view=`*n* keyword in that plot preset definition. Named view presets are selectable using the `View Open` menu command. In this way it is possible to define 1) a global view preset, 2) view presets specific to channel selections defined by plot presets, and 3) user selections of view presets from the `View Open` to override those default selections.

A view preset contains keywords and nested blocks that select analysis displays and channel settings found on the TF32 the `View` menu.

The keyword

`exclude_CHname`=*CHname*

determines the plot presets from which a named view preset is excluded. If a plot preset is selected that excludes certain view presets, by virtue of a view preset referring to channels that are not included in that plot or the inclusion of channels for which a view preset does not apply, those named view entries are grayed out in the `View Open` submenu. The value *CHname* designates a named channel for which a particular view preset does not apply.

For example, the `F0Wnd`, `RMSWnd`, `uglotWnd`, and `integrator` nested blocks specify a channel by number: the keyword `CH=1` selects Channel 1. Suppose a plot preset puts the pressure trace p0 in Channel 1. If we follow one of those nested blocks with the keyword entry `exclude_CHname=p0`, the named view preset in question will be grayed out on the `View Open` menu whenever that plot preset has been selected with `Files New plot` or with `Record`. Please see the file `aero.pre` for examples of such exclusions.

The view preset nested block definitions are delimited in the manner

```
wave_plot:CHname=CHname
end_wave_plot
```

or in the manner

```
F0Wnd
end_F0Wnd
```

The view preset nested block definition headers are as follows

`wave_plot:CHname=`*CHname* enters settings for waveform channels. *CHname* is the channel label (such as the labels `audio`, `p0`, `u0` used by the Aero system); if *Chname* is blank, it refers to all unlabelled channels. The keywords are

- `scale=`s where s is in the range $3 \leq s \leq 14$. Maximum magnification is `scale=3` while minimum magnification is `scale = 14`.
- `ytic=`y where y is spacing between y-axis grid lines on the waveform plot expressed in the units of that waveform.
- `readout=`*readout* where *readout* can be one of `endpoints`, `stats`, or `RMS_dB`.

The `wave_plot` blocks are typically placed in a first, unnamed view preset definition to define default selections for channel scale, tic marks, and numeric readout. Subsequent view presets can use the `preset_readouts` keyword to revert to those default selections.

`F0Wnd` activates a pitch trace display, with keywords

- `CH=`*ch* where *ch* is the channel number of the waveform to be analysed.
- `scale=`*s* where *s* is in the range $3 \leq s \leq 14$. Maximum magnification is `scale=3` while minimum magnification is `scale = 14`.
- `readout=`*readout* where *readout* can be one of `endpoints`, or `stats`.

`RMSWnd` activates an RMS/dB trace display, with keywords

- `CH=`*ch* where *ch* is the channel number of the waveform to be analysed.
- `scale=`*s* where *s* is in the range $3 \leq s \leq 14$. Maximum magnification is `scale=3` while minimum magnification is `scale = 14`.
- `readout=`*readout* where *readout* can be one of `endpoints`, or `stats`.
- `window_ms=`t_w where t_w is the averaging window length for compute RMS.
- `dB` means display dB values.
- `dB_ref=`*dB* where *dB* replaces the 0 dB reference level – used for calibrating the dB scale.

`uglotWnd` activates an LPC inverse filter display, with keywords

- `CH=`*ch* where *ch* is the channel number of the waveform to be analysed.
- `scale=`*s* where *s* is in the range $3 \leq s \leq 14$. Maximum magnification is `scale=3` while minimum magnification is `scale = 14`.
- `readout=`*readout* where *readout* can be one of `endpoints`, `stats`, `pulses`, or `syllables`.
- `mask_tc=`*tc* is the flow mask correction time constant in ms as displayed in an edit box of the `View Open Uglot trace` dialog and described in Section 5.3. The default value of .25 ms corresponds circumferentially-vented mask of 10 sq cm port area.
- `CG_weight=`*CG* is the magnitude weight given to the closed-glottis interval relative to the whole pitch-period interval in the least squares minimization of the LPC calculation used to determine the inverse filter for each pitch period. This values is displayed in an edit box of the `View Open`

`Uglot trace` dialog described in Section 5.3. The default value is 4. A value of 0 corresponds to whole pitch-period least squares only while a value of 100 effectively gives closed-glottis interval least squares only.

`source=`*source* where source is one of `flow_mask`, `hdw_filt`, or `mic`.

`invert=`*tf* where *tf* is `true` or `false`.

`dif=`*tf* where *tf* is `true` or `false`.

`integrator` activates an integrator display. The typical application is to integrate flow to show volume differences – volume derived from integrated flow cannot show absolute volume levels. The integrator is only selectable from the `View Open` menu if it is specified as part of a named view preset. The keywords are

`CH=`*ch* where *ch* is the channel number of the waveform to be analysed.

`scale=`*s* where s is in the range $3 \leq s \leq 14$. Maximum magnification is `scale=3` while minimum magnification is `scale = 14`.

`readout=`*readout* where *readout* can be one of `endpoints`, or `stats`.

`TFWnd` activates a time-frequency spectrogram and optional formant track overlay, with keywords

`CH=`*ch* where *ch* is the channel number of the waveform to be analysed.

`FFTexp=`*e* where e controls the frequency range and in the range $8 \leq s \leq 12$. Maximum magnification is `FFTexp=12` while minimum magnification is `FFTexp = 8`.

`dB_range=`*r* where r controls the dynamic range of the spectrogram display and allowable values are 32, 48, and 64.

`dB_floor=`*f* where f is the minimum displayed dB level: the default is -72.

`bw=`*bw* where *bw* is spectrogram analysis bandwidth in Hz: typical values are 300 for wideband, 45 for narrowband.

`alpha=`α is the preemphasis coefficient where $\alpha = 1$ for preeemphasis, $\alpha = 0$ for no preemphasis.

`LPC` enables the formant track overlay.

`readout=endpoints` activates the formant track numeric readout.

`ReadoutWnd=`*readouttype* activates a numeric readout analysis display in an inset box on the TF32 main window, where *readouttype* is one of

- `CV_resistance` gives automated measurement of glottal resistance from the pressure and flow traces of CV's (consonant-vowel sequences). Glottal resistance is the ratio of peak pressure during the constant divided by mean flow in an interval of steady voicing following the voice onset transient in the following vowel. Pressure and flow measures are automatically corrected for DC offsets by taking flow during the pressure peak and pressure during the sustained flow interval as zero-valued reference points.
- `AC_resistance` gives automated measurement of flow resistance as the change in pressure divided by the change in flow determined by computing a coefficient of correlation of the pressure trace samples with the flow trace samples.
- `CVR_cursor` gives a manual measurement of glottal resistance where the pressure peak is taken as the position of the initial (left) waveform cursor and the sustained flow value is taken as the position of the final (right) waveform cursor.
- `ACR_cursor` gives a manual measurement of flow resistance as the pressure change between the two waveform cursor positions divided by the flow change at those positions.

Keywords include

- `units=`*units* overrides the default units of resistance. For pressure in cm H2O, flow in ml/s, the default units is cm H2O/ml/s while a typical override is `units=ohms` to specify acoustic ohms.
- `units_conversion=`*scalefactor* where *scalefactor* is the number to multiple the ratio of pressure over flow, in their respective units, to obtain resistance in the desired units. For example, `units_conversion=981` is the scale factor required to convert cm H2O/ml/s to acoustic ohms.
- `select_p` specifies the pressure trace. The keyword pair (each keyword on a separate line)

```
select_p
    CHname=p0
```

selects the waveform channel named `p0` in the plot preset as the pressure trace.

`select_u` specifies the flow trace. The keyword pair (each keyword on a separate line)

```
select_u
    CHname=u0
```

selects the waveform channel named `u0` in the plot preset as the flow trace.

These keyword only apply to the automated determination of CV glottal resistance – `ReadoutWnd=CV_resistance`

`show_cursors` shows cursors overlaying the pressure and flow traces showing where the automatic algorithm has located pressure peaks and the sustained flow intervals used in the resistance calculation. This visual feedback is useful in determining when the automatic algorithm has failed to accurately locate these points, which can happen under conditions of plugged pressure tube or a leaky flow mask.

`max_voice_offset=`*offset* limits the interval from the consonant pressure peak to the start of the sustained flow measurement interval to *offset* ms – the default value is 100 ms.

`display=p_min_max` overrides the default display of glottal resistance and show the min and max peak pressure values used in the calculation of glottal resistance.

`display=u_min_max` overrides the default display of glottal resistance and show the min and max sustained voiced flow values used in the calculation of glottal resistance.

7.3.5 Filt preset and automatic data logging

The preset file `aero.pre` contains the filt preset definition block

```
preset:filt
    filter=HPF
        CHname=audio
        bw=50.0
    end_filter
end_preset
```

This definition specifies a high-pass filter with a 50 Hz lower frequency limit for the signal identified by `CHname=audio`; this filter removes the DC artifact from the audio signal prior to displaying the waveform and taking measurements. The high pass filter removes bias when measuring the RMS dB value for that signal.

The menu command `Files New session` specifies automatic logging of waveform recordings into subdirectories keyed by session date and subject/patient identifier number. When automatic logging has been selected, the `Record` command initiates the sequence 1) acquire from the A/D and update real-time displays, 2) when done recording, save the raw waveform recording into files, 3) apply the filter preset to the waveform plots, and 4) append measurements of the filtered waveforms into the file `log.txt` in the session subdirectory. When you retrieve that file with `Files Open` you will get the raw waveform back; you have the option of invoking the `Filter Preset` command to reapply the same conditioning filters used during automatic logging.

It is important to remember that the preset filters are only applied automatically when in the automatic logging mode and only after saving the waveforms. When you retrieve a waveform file with `Files Open`, you are viewing *raw*, unfiltered waveforms and you have to explicitly invoke `Filter Preset` to apply the preset filters used during automatic logging. The reasons for saving raw, unfiltered waveforms to archival storage is that 1) the raw, unfiltered waveform is available if one needs to later change the kind of filtering, and 2) the raw waveform can be used to verify if signals exceeded clipping ranges.

8 Scripting TF32

Back in the days of DOS, batch files provided a way of automating tasks. Command-line file conversion and measurement programs could be applied to large numbers of files. If the particular command-line program did not take a “wild-card” (like `*.wav`), it was not too difficult to put a command line in a `.bat` file for each file you wanted to process, and text editors made even large numbers of files doable. CSpeech provided batch files for command-line waveform playback, waveform file-format conversion, and pitch, jitter, formants, and moments analysis.

While you can still run batch files and DOS programs from the DOS window under Windows, it seems that batch processing went out of style; you were supposed to poke at the menus of a Windows program to do whatever needed to be done.

Windows 98 introduced the Windows Script Host (WSH), a batch language for Windows programs. In conjunction with the Automation features implemented by many Windows applications, including TF32 (requires Lab Automation level), you can write scripts to get Windows applications to do things. In fact, this facility is so powerful, it has become part of a serious social problem. Scripts can be sent as e-mail attachments and invoked from Outlook, and this has become a favorite way to spread computer “virus” programs. That is why it is ever so important to not open e-mail attachments any more if you have not confirmed the source; I will only send you e-mail attachments of TF32 updates or corrections in response to your direct request for your safety in this regard.

TF32 act as an Automation server – requires Basic or Lab Automation level. This means that TF32 is a program that you can activate and command to do useful things from another program. That other program could be written in Delphi Pascal, Visual C++, one of the Windows scripting languages like VBS, or any other programming language from which you are able to write an Automation client, a fancy name for a program that uses Automation to get other programs to do useful things.

An Automation Server has to tell Windows that it is able to be “automated” by other programs. To activate TF32 as an Automation server, go to a DOS prompt and invoke

```
tf32 /regserver
```

This action makes an entry in the Windows Registry, a database that tells Windows what programs and resources are available on your computer. To remove TF32 from the Registry and deactivate it as an Automation server, invoke

```
tf32 /unregserver
```

If TF32 has been activated as an Automation server, a Registry entry tells Windows about the existence of the Automation object `TF32.Application`. Automation objects have “verbs”, commands that result in some action and they have “properties”, object variables that you can read and write.

The `TF32.Application` object implements the verbs

`ShowWindowMaximized` tells TF32 to maximize its main window. ShowWindowMaximized takes no arguments.

`Open` opens a waveform, a wave list (list of entries made up of a start time, and end time, and a waveform file for which those times apply), or a preset file (configures the TF32 display). Open takes one argument, the file name.

`ExportWave` saves the open waveform. It takes 2 arguments: the file name to save the waveform followed by a numeric code for the file format. Formats are 0) CSpeech/TF32 1) CSpeech/TF32 lossless compressed, 2) Microsoft .WAV (RIFF format), 3) SPHERE (used by the TIMIT speech database), and 4) headerless binary.

`MeasureSequence` invokes the `Edit Measure sequence` command on the TF32 menu. This steps through the entries of an open wave list or it opens wave files that follow in alphabetical order the open wave file. For each wave file, it saves the numeric readouts for waveforms and waveform analysers to a file. MeasureSequence takes the ASCII data file name for saving numeric values as a single argument. It appends measurements to that file; you need to erase that file first if you want to overwrite.

`CloseAnalysers` closes any open waveform analysers (pitch trace, RMS, uglot) in the TF32 display. It takes no arguments.

`CloseTFs` closes any open time-frequency displays. It takes no arguments.

`SaveAnalysis` saves the indicated analysis trace to a file. It takes 3 arguments: ianalysis: the index, starting at 1, of the analysis trace (pitch, RMS, uglot) from the TF32 display to save; the file name in which to output the analysis trace; the format of that file. Formats 0-4 save the analysis trace as a binary waveform in the formats specified above for `ExportWave`.

Format 5 saves the pitch trace as ASCII text with the first column the times (ms) of glottal epochs and the second column the pitch in Hz. Format 5 also saves the uglot (inverse filter glottal airflow) glottal parameters as ASCII text (a description of the file format follows).

`SaveFormants` saves the formant traces, if activated, from the indicated time-frequency display to a file. It takes 2 arguments: the index, starting at 1, of the time-frequency display, and the file name in which to output formants. The file is ASCII text in the same format as the `.FBW` file output by the `Save` button on the time-frequency dialog (see Section 5.4.3).

`Play` outputs the open waveform to the sound card; it takes no arguments.

`HPF` applies a zero-phase high pass filter to a waveform channel. It takes 3 arguments: ch: the waveform channel number in the TF32 display starting with 1, 0 means whatever channel is selected by the channel clicker; bw: a numeric value for the filter bandwidth in Hz (50 is a typical value for high-pass filtering to get rid of low-frequency artifact); and newchan: True to filter into a new waveform channel, False to overwrite the channel being filtered.

Each of the above verbs return a Boolean (True/False) value where True denotes successful completion of the operation.

The `TF32.Application` object implements the properties

`Visible` is the visibility state of the TF32 main window. Set `Visible` to True if you want to show TF32 on the screen. Set `Visible` to False if you want to use TF32 for processing without it appearing on the screen. Visible is initially set to False. If you leave Visible with the value False when you close the TF32.Application object, TF32 will close down. If you leave Visible set to True, TF32 stays on the screen until you close TF32 by selecting `Files Exit` from the TF32 menu.

`NChan` is the number of waveform channels in the TF32 display. This property is read-only.

`NAnalysis` is the number of analysis traces (pitch, RMS, uglot) in the TF32 display. This property is read-only.

`NTFWnd` is the number of time-frequencies displays active in TF32. This property is read-only.

`YExp` is an array property with integer index ranging from 1 to the value of `NChan`. This property is the channel magnification, ranging from 3 (max magnification) to 14 (min magnification), of the indexed waveform channel.

`YTic` is an array property with integer index ranging from 1 to the value of `NChan`. This property gives the spacing between vertical grid lines, in calibrated units, of the indexed waveform channel. A value of zero means no grid lines.

`ReadoutName` is an array property with integer index ranging from 1 to the vale of `NChan`. This property is a text string that names the numeric analyser for the indexed waveform channel. Valid values are `none`, `endpoints`, `stats`, and `RMS_dB`, corresponding to selections from the `View Wave plots` dialog.

The following verbs of the `TF32.Application` object return other object references that implement additional verbs.

`CreateViewOpen` creates a new ViewOpen object and returns an Automation reference to that object. The ViewOpen object allows specifying keywords that define analyser settings and adding analysers (pitch, RMS, uglot, time-frequency) with those settings to the TF32 display. Multiple invocations of `CreateViewOpen` will create multiple ViewOpen objects; this is helpful when you want to maintain multiple sets of analyser settings for reopening analysers with those settings.

`CreateCalibrator` creates a new Calibrator object and returns an Automation reference to that object. The Calibrator object can take measurements of waveform channels in the display to determine the calibration range and null offset correction for waveforms such as pressure and flow, and it can apply those corrections to waveforms input into the display with the `Open` verb. Multiple invocations of `CreateCalibrator` create multiple Calibrator objects; this allows using a separate Calibrator object for each channel to calibrate.

The ViewOpen object (created with the `CreateViewOpen` verb of the `TF32.Application` object) implements the verb

`AddParamStr` takes 1 argument, a text string to be added to a list of keyword entries used in opening an analyser in the TF32 display in the manner of the `View Open` menu command from TF32.

This verb return a Boolean True to indicated success. The idea is to add keyword entries that specify properties of an analyser – target waveform channel, analyser settings – and then open an analyser using those settings by invoking one of the following verbs. The keywords specific to each type of analyser are described in Section 7.3.4.

Each of the following verbs takes no arguments and returns the integer index, starting at 1, of that analyser in the TF32 display.

`OpenPitch` opens a pitch trace analyser.

`OpenRMS` opens an RMS trace analyser.

`OpenUglot` opens a uglot analyser – LPC glottal inverse filter analysis and glottal epoch and parameter extraction from either a wideband flow mask or a microphone signal.

`OpenTF` opens a time-frequency display.

The `SaveAnalysis` and `SaveFormants` verbs of `TF32.Application` use the returned integer index to specify which analysis trace for formant values to save.

The glottal parameters extracted by the uglot analyser can be saved to an ASCII file by applying the `SaveAnalysis` verb to the analyser index returned by `OpenUglot` and by specifying format 5. The format of each line is

epoch time (ms)
pitch period (ms)
OQ (open quotient – 0.0 to 1.0)
CG (closed-glottis flow)
avg (average flow for complete pitch period cycle)
peak (peak flow)
mFD (maximum flow declination -
if calibrated for cc/s flow it is expressed as cc/s/ms,
if calibrated for l/s flow it is expressed as l/s/s)

The epoch time is the start of the open phase of a glottal cycle. The start of the closed phase is epoch_time + OQ*pitch_period.

For microphone inverse filtering, the average and peak flow are given relative to CG flow (because CG flow is arbitrary for an AC-coupled mic signal. The line format becomes

epoch time (ms)
pitch period (ms)
OQ (open quotient – 0.0 to 1.0)
avg - CG
peak - CG
mFD (maximum flow declination)

The Calibrator object (created with the `CreateCalibrator` verb of the `TF32.Application` object) implements these verbs that return Boolean True indicating success.

`Null` takes one argument, the channel number of a waveform channel. Null computes the average of the waveform in that channel and assigns it the null or zero value. Null is usually applied to a channel of an opened waveform file into which a null level had been previously saved.

`Cal` takes two arguments, the channel number of a waveform channel and a floating-point numeric value to be assigned to that channel. Cal computes the average of the waveform in that channel and assigns it the floating-point numeric value as the average level of that channel in calibrated units. Cal is usually applied after Null, and it is applied to a channel of an opened waveform file into which a known, calibrated level has been previously saved.

`ApplyCal` takes a waveform channel number as its argument. It applies the range and offset determined by the Null and Cal procedure (and Units property – see description that follows) and applies it to the designated waveform channel. Open the waveform you want to calibrate and invoke `ApplyCal` on the channel containing that waveform.

Calibrator implements these properties

`Units` is a text string giving the units name (such as cmH2O for pressure or ml/s for flow). Set the units to the correct value before invoking the `ApplyCal` verb.

`Range` is the peak-to-peak data range in calibrated units for the 15-bit resolution sample values in a TF32 waveform channel. Range is read-only and is updated by performing the Null and Cal procedures described above.

`Offset` is the offset correction in calibrated units that is subtracted from range-corrected sample values to obtain calibrated sample values. Offset is read-only and is updated by Null and Cal.

The sequence is to `Open` the null waveform, invoke `Null` on that waveform, `Open` the cal waveform, invoke `Cal` on that waveform, set the `Units` property to the correct units name, and finally `Open` a waveform to be calibrated and invoke `ApplyCal`. If you want to save the calibrated waveform with `ExportWave`, you need to select formats 0 or 1 to save the calibration information in the file header. If you export the waveform to Microsoft WAV (RIFF format – format 2), you will lose that calibration setting. Is there a way to open a RIFF waveform in a scripting language and apply the calibration to the raw integer samples by writing a program in that scripting language?

The Calibrator also implements the following verb that returns a floating-point numeric value useful in performing the calibration correction to raw integer sample values as input from a binary file.

`GetScale` takes an integer res value as its single argument. It returns a scale factor to multiply an integer sample at the specified res (bits resolution) to return a calibrated value.

For example, to correct a raw integer sample stored in 16-bit resolution, such as from a Microsoft WAV (RIFF format) file compute

cal_val = (raw_val * Cal.GetScale(16)) - Cal.Offset

where Cal is a variable containing a reference to the Calibrator object, Cal.GetScale(16) invokes the `GetScale` verb and `Cal.Offset` returns the value of the offset property.

8.1 Example scripts

The following examples are written in Visual Basic Script (`.vbs` file extension). Simply click on a `.vbs` file in Explorer or from the Desktop to run that script.

Play a sequence of two waveforms, running TF32 in the background.

```
Dim RunTF32, success
Set RunTF32 = CreateObject("TF32.Application")
success = RunTF32.Open("c:\csp\jw12\tp010.acc")
success = RunTF32.Play
success = RunTF32.Open("c:\csp\jw16\tp010.acc")
success = RunTF32.Play
Set RunTF32 = Nothing
```

The line

```
Dim RunTF32, success
```

declares 2 variables. Variable `RunTF32` will hold the TF32 Automation object; variable `success` will hold the return value every time we invoke an Automation verb.
The line

```
Set RunTF32 = CreateObject("TF32.Application")
```

creates the TF32 Automation object and assigns it to the variable `RunTF32`. The line at the very end

```
Set RunTF32 = Nothing
```

clears the RunTF32 variable, insuring that the TF32 application object is released when this script ends.

This last line is not absolutely necessary because Windows will free the TF32 application object when the script is done, but it is a good programming habit (will keep you out of trouble with more complicated scripts) to clean up after yourself rather than having someone clean up after you. Also, releasing the TF32 application object only effects that object; it may not close down the TF32 program depending on whether the TF32 is visible or not. In this script, the TF32 program closes down because the Visible property was never set to True.
Each pair of lines

```
success = RunTF32.Open("c:\csp\jw12\tp010.acc")
success = RunTF32.Play
```

tells TF32 to open the named waveform file and play it through the sound card.

The dot notation as in

```
success = RunTF32.Open("c:\csp\jw12\tp010.acc")
```

is an example of object-oriented programming. Just as nothing gets done in the workplace without a lot of e-mail messages, object-oriented programming can be thought of sending e-mail messages to software programs that know what to do when they get those messages. The variable `RunTF32` holds a reference (like an e-mail address) to the TF32 Automation object called `TF32.Application`. `RunTF32.Open` means apply the Open verb or action to that object (like sending an e-mail with the "Open" subject heading): it instructs the TF32 Automation object to go tell TF32 to open a file.

Open a waveform, apply a 50 Hz high-pass filter, and save that waveform to a Microsoft .WAV (RIFF format) file in the root directory of the C: drive.

```
Dim RunTF32, success
Set RunTF32 = CreateObject("TF32.Application")
success = RunTF32.Open("c:\csp\jw12\tp010.acc")
success = RunTF32.HPF(1,50,False)
success = RunTF32.ExportWave("c:\tp010.wav",2)
Set RunTF32 = Nothing
```

This script has the same first 2 and last lines as before that create and release the TF32 application object. This script uses the Open verb to input a waveform, the HPF verb to filter it, and the ExportWave verb to save the filtered waveform in another file. The argument to ExportWave with the value of 2 tells TF32 to save the waveform in Microsoft .WAV (RIFF) formant.

Make TF32 visible on the screen, maximize the main window, open a waveform, and leave TF32 open.

```
Dim RunTF32, success
Set RunTF32 = CreateObject("TF32.Application")
RunTF32.Visible = True
success = RunTF32.ShowWindowMaximized
success = RunTF32.Open("c:\csp\jw12\tp010.acc")
Set RunTF32 = Nothing
```

The line

```
RunTF32.Visible = True
```

sets a *property* of the TF32 Automation object referenced by the `RunTF32` variable of the script. Since Visible is left set to True when the script ends, TF32 will be left up on the screen showing the opened waveform.

TF32 has a batch-mode built in: the wave list file and the `Edit Measure sequence` command. The wave list file is made up of entries where each entry is an initial and final time followed by a waveform file name. If you Open a wave list, activate the numeric readouts you want, apply filters to any waveform channels if needed, and invoke `Edit Measure sequence`, TF32 will open each entry on that list, update the numeric readouts, and save readout values to a text file.

What if you have a large number of wave lists? TF32 allows you to automate the above steps from a script file.

Show TF32 on the screen, open a wave list, set the waveform readout to dB, configure and a open pitch trace and a time frequency display with formant readout, measure each entry in the wave list and save numeric readout values to the file `meas.dat`, and close TF32 upon completion.

```
Dim RunTF32, success, F0Open, TFOpen, index
Set RunTF32 = CreateObject("TF32.Application")
RunTF32.Visible = True
success = RunTF32.Open("c:\csp\jw12\tf32.lst")
RunTF32.ReadoutName(1) = "RMS_dB"
Set F0Open = RunTF32.CreateViewOpen
success = F0Open.AddParamStr("CH=1")
success = F0Open.AddParamStr("scale=12")
success = F0Open.AddParamStr("readout=endpoints")
index = F0Open.OpenPitch
success = RunTF32.CloseTFs
Set TFOpen = RunTF32.CreateViewOpen
success = TFOpen.AddParamStr("CH=1")
success = TFOpen.AddParamStr("bw=300")
success = TFOpen.AddParamStr("FFTexp=10")
success = TFOpen.AddParamStr("LPC")
success = TFOpen.AddParamStr("readout=endpoints")
index = TFOpen.OpenTF
success = RunTF32.HPF(1,50,False)
success = RunTF32.MeasureSequence("c:\csp\jw12\meas.dat")
RunTF32.Visible = False
Set RunTF32 = Nothing
Set F0Open = Nothing
Set TFOpen = Nothing
```

The start and finish of the script is much as before, only the Visible property is set to False so that TF32 closes when the script is done.

The lines

```
Set F0Open = RunTF32.CreateViewOpen
success = F0Open.AddParamStr("CH=1")
success = F0Open.AddParamStr("scale=12")
success = F0Open.AddParamStr("readout=endpoints")
index = F0Open.OpenPitch
```

configure and open the pitch trace. The line

```
success = RunTF32.CloseTFs
```

closes the default time-frequency display before configuring and opening a time-frequency display with desired settings.

Show TF32 on the screen, calibrate using the wave file `null.wav` for null reference and `cal.wav` for a 100.0 ml/s calibrated reference, open the wave file `pa1210.wav`, configure and open a uglot trace to show the LPC inverse filter glottal waveform, and save the inverse filter waveform in Microsoft WAV (RIFF format) to `pa1210.flt` and the glottal parameters in ASCII to `pa1210.glo`.

```
Dim RunTF32, Cal, ViewOpen, success, iglot
Set RunTF32 = CreateObject("TF32.Application")
RunTF32.Visible = True
' perform the calibration
Set Cal = RunTF32.CreateCalibrator
success = RunTF32.Open("null.wav")
success = Cal.Null(1)
success = RunTF32.Open("cal.wav")
success = Cal.Cal(1,100.0)
Cal.Units = "ml/s"
success = RunTF32.Open("pa1210.wav")
success = Cal.ApplyCal(1)
' configure and open an inverse filter
Set ViewOpen = RunTF32.CreateViewOpen
success = ViewOpen.AddParamStr("CH=1")
success = ViewOpen.AddParamStr("mask_tc=0.25")
success = ViewOpen.AddParamStr("CG_weight=4.0")
success = ViewOpen.AddParamStr("readout=syllables")
success = ViewOpen.AddParamStr("source=flow_mask")
iglot   = ViewOpen.OpenUglot ' returns the index number of the analyser
' save the inverse filter as a waveform trace and as parameters
success = RunTF32.SaveAnalysis(iglot, "pa1210.flt", 2)
success = RunTF32.SaveAnalysis(iglot, "pa1210.glo", 5)
Set Cal = Nothing
Set ViewOpen = Nothing
Set RunTF32 = Nothing
```