Audio data acquisition also stops automatically when an experiment is closed.

See Also

The **SoundInStartChart** and **SoundInStatus** operations.

SoundLoadWave

`SoundLoadWave [flags] waveName [ ,fileNameStr ]`

The SoundLoadWave operation loads sound data from the named file into a wave. Mono, stereo, surround-sound, and high-resolution sound formats are supported.

The SoundLoadWave operation was added in Igor Pro 7.00.

Parameters

waveName is the name of the wave to load the sound into.

If *fileNameStr* is omitted or is "", SoundLoadWave displays an Open File dialog.

The file to be loaded is specified by *fileNameStr* and /P=*pathName* where *pathName* is the name of an Igor symbolic path. *fileNameStr* can be a full path to the file, in which case /P is not needed, a partial path relative to the folder associated with pathName, or the name of a file in the folder associated with pathName. If SoundLoadWave can not determine the location of the file from *fileNameStr* and *pathName*, it displays a dialog allowing you to choose the file.

If you use a full or partial path for *fileNameStr*, see **Path Separators** on page III-451 for details on forming the path.

Flags

/I [=*filterStr*]	Force interactive mode. Use optional filter string to limit allowable file extensions. See **Open File Dialog File Filters** on page IV-149.
/O	Overwrite existing waves in case of a name conflict.
/P=*pathName*	Specifies the folder to load the file from. *pathName* is the name of an Igor symbolic path, created via **NewPath**. It is not a file system path like "`hd:Folder1:`" or "`C:\\Folder1\`". See **Symbolic Paths** on page II-22 for details.
/Q	Quiet: Doesn't print message to history area, and doesn't abort, if the sound can not be loaded. V_Error is set to the returned error code, which will be zero if there was no error.
/S=(*startT*,*endT*)	Load a subrange of the sound resource. *startT* and *endT* are in seconds, clipped to the duration of the loaded sound.
/TMOT= *timeOut*	Aborts load if *timeOut*, in seconds, is exceeded.

Details

SoundLoadWave uses Core Audio on Macintosh and Qt framework calls on Windows. Note that some files can not be loaded due to digital rights managment issues even though they can be played.

If *waveName* specifies a wave that does not exist, it is created. The wave is redimensioned to a wave type that maintains the numeric precision of the sound data. If the wave can not be created or resized to fit the loaded data then SoundLoadWave returns an error.

If *waveName* does exist, the wave is overwritten only if the /O flag is specified. Without the /O flag SoundLoadWave returns an error.

Multi-channel audio is loaded into sequential columns of the destination wave.

On Macintosh, Igor uses Core Audio to produce 32-bit or 64-bit floating point waves. The BITS value in S_info, described below, may be zero for some formats.

On Windows, SoundLoadWave uses the smallest Igor wave data type that preserves the number of bits in the audio. Igor doesn't have a 24-bit data type, so these values are stored in a 32-bit integer wave.

Output Variables

SoundLoadWave sets these output variables:

V_flag	Set to 1 if a sound is loaded and fits into available memory, 0 otherwise.
V_Error	Set if /Q is specified, V_Error is set to a non-zero error code if something went wrong or to zero on success. Negative returned codes are system-dependent, positive are Igor-defined errors. V_Error = 1 means there wasn't enough memory to load the (uncompressed) sound.
S_path	Set to the full file path of the loaded file, not including the file name.
S_fileName	Set to the name of the loaded file.
S_waveNames	Set to the name of loaded wave.
S_info	Information about the loaded sound.

If the sound file exists, SoundLoadWave sets the string variable S_info to:

```
"FILE:nameOfFile;FORMAT:soundFileFormat;CHANNELS:numChannels;CHANNEL_LAYOUT:ch
annelLayoutDescription;CHANNEL_ORDER:channelsList;BITS:numBits;SAMPLES:numSamp
les;RATE:samplesPerSec;"
```

The *soundFileFormat* and *channelLayoutDescription* values are text descriptions of the sound data in the file, and are written in the localized language. This information is available only on Macintosh and may or may not be present in a given sound file.

The *channelsList* value is a comma-separated list of channel names, always in English abbreviations, such as "L,R" or "L,R,C,LFE,Ls,Rs". The meaning of the abbrevations:

channelList Abbreviation	Channel or Speaker Names
L	Front Left
R	Front Right
C	Front Center
LFE	Low Frequency Effects
Ls	Left Surround (Back Left)
Rs	Right Surround (Back Right)
Lc	Left Center (Front Left of Center)
Rc	Right Center (Front Right of Center)
Cs	Center Surround (Back Center)
Lsd	Left Surround Direct (Side Left)
Rsd	Right Surround Direct (Side Right)
Ts	Top Center Surround (Top Center)
Vhl	Vertical Height Left (Top Front Left)
Vhc	Vertical Height Center (Top Front Center)
Vhr	Vertical Height Right (Top Front Right)
Rls	Rear Left Surround (Top Back Left)
Rcs	Rear Center Surround (Top Back Center)
Rrs	Rear Right Surround (Top Back Right)