

"Igor"	Loads data from Igor Pro application.
--------	---------------------------------------

If you specify */P=pathName*, note that it 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.

There are no sounds in the Igor Pro application file.

If the file is not fully specified and *fileNameStr* is not one of these special values, then PlaySnd presents a dialog from which you can select a file. "Fully specified" means that Igor can determine the name of the file (from the *fileNameStr* parameter) and the folder containing the file (from the */P=pathName* flag or from the *fileNameStr* parameter).

PlaySnd sets the variable *V\_flag* to 1 if the sound exists and fits in available memory or to 0 otherwise.

If the sound exists, PlaySnd also sets the string variable *S\_Info* to:

```
"SOURCE:sourceName; RESOURCENAME:resourceName; RESOURCEID:resourceID"
```

If the sound is not a resource then *resourceName* is "" and *resourceID* is 0. *sourceName* will be the name of the file that was loaded or "Clipboard", "System" or "Igor".

### Examples

```
PlaySnd/I=1/P=mySnds/Z "Wild Eep"
If (V_flag) // Any 'snd ' in the "Wild Eep" file?
    Print S_info // Yes, print resource number, etc.
Endif
```

This prints the following into the history area:

```
SOURCE:resource fork; RESOURCENAME:Wild Eep; RESOURCEID:8;
```

## PlaySound

**PlaySound** [*/A[=a]* */BITS=bits* */C*] *soundWave*

**PlaySound** */A[=a]* */BITS=bits* {*soundWave1*, *soundWave2* [, *soundWaveN...*] }

The PlaySound operation plays the audio samples in the named wave. The various sound output parameters — number of samples, sample rate, number of channels, and number of bits of resolution — are determined by the corresponding parameters of the wave.

### Flags

*/A[=a]* Plays sounds asynchronously so that sounds will continue to play after the command itself has executed.

*/A=0:* Same as no */A* flag.

*/A=1:* Plays sounds asynchronously; same as */A*.

*/A=2:* Stop playing any current sound before starting this one.

*/A=3:* Return with user abort error if output buffers are full (rather than waiting.)  
Use `GetRTErr(1)` to detect and clear the error condition.

*/BITS=bits* Controls the number of bits used for each sound sample sent to the sound output hardware.  
Use */BITS=24* with a 32-bit integer wave for 24-bit sound data capable of representing values from -8,388,608 to +8,388,607.

If you omit */BITS* or use */BITS=0*, PlaySound uses the wave's data type and size to determine how many bits are used for each sound sample.

The */BITS* flag was added in Igor Pro 9.00.

*/C* Obsolete - do not use.

On Windows */C* causes sound wave data greater than 16-bits to be converted to 16-bit integer. Such data should range from -32768 to +32767.

On Macintosh */C* is ignored.