/WINF=*windowKind*

Premultiplies a data seggment with the selected window function. The default window is Hanning. See **Window Functions** on page V-225 for details.

/Z Ignores errors. V_flag is set to -1 for any error and to zero otherwise.

Details

The Short-Time Fourier Transform is a time-frequency representation for a 1D array. The squared magnitude of the transform is known as the "spectrogram" for time series or "sonogram" in the case of sound input. The operation comprises the following steps:

1. Sampling the data.

   A segment of size specified by /SEGS is sampled from *srcWave* centered about the first point in the wave or as specified by /RX or /RP flags. When the first point is at the beginning of the wave the centering implies that the first half of the segment is set to zero. If the first point is somewhere else the operation uses as many points as are available in the wave and sets the rest to zero. End effects are mitigated by scaling the result by a factor that accounts for the actual number of source data used in the segment. Subsequent segments are each centered at *hopSize* points from the previous center.

2. Apply windowing.

   The selected segment is multiplied by the specified window function.

3. Apply padding.

   If padding is specified the windowed data are centered onto a zero padded array. Padding may be used to simulate longer arrays for improved spectral resolution.

4. Compute the FFT.

   Initial complex transform may be further processed according to the /OUT flag.

See Also

Fourier Transforms on page III-270, **FFT**, **CWT**, **WignerTransform**, **DSPPeriodogram**, **LombPeriodogram**

StopMSTimer

StopMSTimer(*timerRefNum*)

The StopMSTimer function frees up the timer associated with the *timerRefNum* and returns the number of elapsed microseconds since StartMSTimer was called for this timer.

Parameters

timerRefNum is the value returned by StartMSTimer or the special values -1 or -2. If *timerRefNum* is not valid then StopMSTimer returns 0.

On Windows, passing -1 returns the clock frequency of the timer and. On Macintosh, it returns NaN.

Passing -2 returns the time in microseconds since the computer was started.

Details

If you want to make sure that all timers are free, call StopMSTimer ten times with *timerRefNum* equal to 0 through 9. It is OK to stop a timer that you never started.

The function result may exclude the time the system has spent in sleep states. As of this writing this applies to Macintosh only, but this behavior may change in the future since it is determined by the operating system.

Examples

How long does an empty loop take on your computer?

```
Function TestMSTimer()
   Variable timerRefNum
   Variable microSeconds
   Variable n
```