Keyword	Information Following Keyword
NAME	Name of the guide.
WIN	Name of the window or subwindow containing the guide.
TYPE	The value associated with this keyword is either *User* or *Builtin*. A *User* type denotes a guide created by the DefineGuide operation, equivalent to dragging a new guide from an existing one.
HORIZONTAL	Either 0 for a vertical guide, or 1 for a horizontal guide.
POSITION	The position of the guide in points. This is the actual position relative to the left or top edge of the window, not the relative position specified to DefineGuide.

Keyword	Information Following Keyword
GUIDE1	The guide is positioned relative to GUIDE1.
GUIDE2	In some cases, the guide is positioned at a fractional position between GUIDE1 and GUIDE2. If the guide does not use GUIDE2, the value will be "".
RELPOSITION	The position relative to GUIDE1 (and GUIDE2 if applicable). This is the same as the *val* parameter in DefineGuide. The returned value is in units of points if only GUIDE1 is used, or a fractional value if both GUIDE1 and GUIDE2 are used.

GuideNameList

GuideNameList(*winNameStr, optionsStr*)

The GuideNameList function returns a string containing a semicolon-separated list of guide names from the named host window or subwindow.

Parameters

winNameStr can be "" to refer to the top host window.

When identifying a subwindow with *winNameStr*, see **Subwindow Syntax** on page III-92 for details on forming the window hierarchy.

optionsStr is used to further qualify the list of guides. It is a string containing keyword-value pairs separated by commas. Use "" to list all guides. Available options are:

TYPE:type	type = BuiltIn: List only built-in guides.
	type = User: List only user-defined guides, those created by the DefineGuide operation or by manually dragging a new guide from an existing one.
HORIZONTAL:h	h = 0: List only non-horizontal (that is, vertical) guides.
	h = 1: List only horizontal guides.

Example

```
String list = GuideNameList("Graph0", "TYPE:Builtin,HORIZONTAL:1")
```

See Also

The **DefineGuide** operation and the **GuideInfo** function.

Hanning

Hanning *waveName* [, *waveName*]…

Note: The **WindowFunction** operation has replaced the Hanning operation.

The Hanning operation multiplies the named waves by a Hanning window (which is a raised cosine function).

You can use Hanning in preparation for performing an FFT on a wave if the wave is not an integral number of cycles long.