

**See Also**

Another command related to contour plots and waves: **ContourNameToWaveRef**.

For commands referencing other waves in a graph: **TraceNameList**, **WaveRefIndexed**, **XWaveRefFromTrace**, **TraceNameToWaveRef**, **CsrWaveRef**, **CsrXWaveRef**, **ImageNameList**, and **ImageNameToWaveRef**.

## ContourNameToWaveRef

**ContourNameToWaveRef** (*graphNameStr*, *contourNameStr*)

Returns a wave reference to the wave corresponding to the given contour name in the graph window or subwindow named by *graphNameStr*.

**Parameters**

*graphNameStr* can be "" to refer to the top graph window.

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

The contour name is identified by the string in *contourNameStr*, which could be a string determined using ContourNameList. Note that the same contour name can refer to different waves in different graphs, if the waves are in different data folders.

**See Also**

The **ContourNameList** function.

For a discussion of wave reference functions, see **Wave Reference Functions** on page IV-197.

## ContourZ

**ContourZ** (*graphNameStr*, *contourInstanceNameStr*, *x*, *y* [,*pointFindingTolerance*])

The ContourZ function returns the interpolated Z value of the named contour plot data displayed in the named graph.

For gridded contour data, ContourZ returns the bilinear interpolation of the four surrounding XYZ values.

For XYZ triplet contour data, ContourZ returns the value interpolated from the three surrounding XYZ values identified by the Delaunay triangulation.

**Parameters**

*graphNameStr* can be "" to specify the topmost graph.

*contourNameStr* is a string containing either the name of the wave displayed as a contour plot in the named graph, or a contour instance name (wave name with "#n" appended to distinguish the nth contour plot of the wave in the graph). You might get a contour instance name from the **ContourNameList** function.

If *contourNameStr* contains a wave name, *instance* identifies which contour plot of *contourNameStr* you want information about. *instance* is usually 0 because there is normally only one instance of a wave displayed as a contour plot in a graph. Set *instance* to 1 for information about the second contour plot of *contourNameStr*, etc. If *contourNameStr* is "", then information is returned on the *instanceth* contour plot in the graph.

If *contourNameStr* contains an instance name, and *instance* is zero, the instance is taken from *contourNameStr*. If *instance* is greater than zero, the wave name is extracted from *contourNameStr*, and information is returned concerning the *instanceth* instance of the wave.

*x* and *y* specify the X and Y coordinates of the value to be returned. This may or may not be the location of a data point in the wave selected by *contourNameStr* and *instance*.

Set *pointFindingTolerance* =1e-5 to overcome the effects of perturbation (see the perturbation keyword of the **ModifyContour** operation).

The default value is 1e-15 to account for rounding errors created by the triangulation scaling (see **ModifyContour**'s **equalVoronoiDistances** keyword), which works well **ModifyContour** perturbation=0.

A value of 0 would require an exact match between the scaled x/y coordinate and the scaled and possibly perturbed coordinates to return the original z value; that is an unlikely outcome.