

Chapter IV-3 — User-Defined Functions

```
if (strlen(graphName)==0 || WinType(graphName) !=1)
    return $"
endif

// Make a wave to contain wave references
Make /FREE /WAVE /N=0 listWave

Variable index = 0
do
    // Get wave reference for next Y wave in graph
    WAVE/Z w = WaveRefIndexed(graphName,index,1)
    if (WaveExists(w) == 0)
        break                // No more waves
    endif

    // Add wave reference to list
    InsertPoints index, 1, listWave
    listWave[index] = w

    index += 1
while(1)                      // Loop till break above

return listWave
End
```

The returned wave reference wave is a free wave. See **Free Waves** on page IV-91 for details.

For an example using a wave reference wave for multiprocessing, see **Wave Reference MultiThread Example** on page IV-327.

Data Folder References

The data folder reference is a lightweight object that refers to a data folder, analogous to the wave reference which refers to a wave. You can think of a data folder reference as an identification number that Igor uses to identify a particular data folder.

Data folder reference variables (DFREFs) hold data folder references. They can be created as local variables, passed as parameters and returned as function results.

The most common use for a data folder reference is to save and restore the current data folder during the execution of a function:

```
Function Test()
    DFREF saveDFR = GetDataFolderDFR()    // Get reference to current data folder

    NewDataFolder/O/S MyNewDataFolder      // Create a new data folder
    . . .
    SetDataFolder saveDFR                // Restore current data folder
End
```

Data folder references can be used in commands where Igor accepts a data folder path. For example, this function shows three equivalent methods of accessing waves in a specific data folder:

```
Function Test()
    // Method 1: Using paths
    Display root:Packages:'My Package':yWave vs root:Packages:'My Package':xWave

    // Method 2: Using the current data folder
    DFREF dfSave = GetDataFolderDFR() // Save the current data folder
    SetDataFolder root:Packages:'My Package'
```