```
   endif
   SetDrawEnv fillpat= 0       // polys are not filled

   if( s.marker == 0 )         // 90 deg U open to the right
      DrawPoly s.x+size,s.y-size,1,1,{size,-size,-size,-size,-size,size,size,size}
   elseif( s.marker == 1 )     // 90 deg U open to the left
      DrawPoly s.x-size,s.y-size,1,1,{-size,-size,size,-size,size,size,-size,size}
   elseif( s.marker == 2 )     // Cap Gamma
      DrawPoly s.x+size,s.y-size,1,1,{size,-size,-size,-size,-size,size}
   elseif( s.marker == 3 )     // Cap Gamma reversed
      DrawPoly s.x-size,s.y-size,1,1,{-size,-size,size,-size,size,size}
   endif
   return 1
End

Window Graph1() : Graph
   PauseUpdate; Silent 1       // building window...
   Make/O/N=10 testw=sin(x)
   Display /W=(35,44,430,252) testw,testw,testw,testw
   ModifyGraph offset(testw#1)={0,-0.2},offset(testw#2)={0,-0.4},
               offset(testw#3)={0,-0.6}
   ModifyGraph mode=3,marker(testw)=100,marker(testw#1)=101,marker(testw#2)=102,
               marker(testw#3)=103
   SetWindow kwTopWin,markerHook={AudiologyMarkerProc,100,103}
EndMacro
```

See also the Custom Markers Demo experiment - in Igor choose File→Example Experiments→Feature Demos 2→Custom Markers Demo.

Tooltip Hook Functions

Igor displays tooltips for traces and images in graphs if you have selected Graph→Show Trace Info Tags and for table data cells if you have selected Table→Show Column Info Tags. You can also set help text for user-defined controls. In Igor Pro 9.00 or later, you can customize those tooltips by creating a tooltip hook function and activating it for a graph or control panel using **SetWindow**.

A tooltip hook function is similar to a window hook function (see **Named Window Hook Functions** on page IV-295). Igor calls the tooltip hook function for a graph when the mouse hovers over a trace or image, and for a table when the mouse hovers over a cell associated with a wave. The tooltip hook function is also called for a graph or control panel when the mouse hovers over a control.

Unlike a window hook function, you can associate a tooltip hook function with either a top-level graph or control panel window or with a control panel subwindow.

A tooltip hook function takes one parameter - a **WMTooltipHookStruct** structure. The structure contains fields describing the trace, image, wave or control for which tooltip help is being requested and fields that allow you to return information to Igor.

When Igor calls a tooltip hook function, the tooltip field or the structure is preset to the text that Igor has composed for the tooltip. The function can alter it, add to it, or replace it completely. If you want to specify the displayed tooltip, set the tooltip field to your desired text and return 1 from the hook function. Otherwise return 0. Other return values are reserved for future use.

Here is a tooltip hook function that generates tool tips for traces in graphs:

```
Function MyGraphTraceTooltipHook(s)
   STRUCT WMTooltipHookStruct &s

   Variable hookResult = 0     // 0 tells Igor to use the standard tooltip

   // traceName is set only for graphs and only if the mouse hovered near a trace
   if (strlen(s.traceName) > 0)
      hookResult = 1           // 1 tells Igor to use our custom tooltip
      WAVE w = s.yWave         // The trace's Y wave
```