Your function can use the DrawXXX operations to draw the marker. The function is called each time the marker is drawn and should not do anything other than drawing the marker. The function should return 1 if it handled the marker or 0 if not.

Because the user-defined function runs during a drawing operation that cannot be interrupted without crashing Igor, the debugger cannot be invoked while it is running. Consequently breakpoints set in the function are ignored. Use **Debugging With Print Statements** on page IV-212 instead.

The marker number range, which you specify via the SetWindow markerHook call, can be any positive integers less than 1000 and can overlap built-in marker numbers.

WMMarkerHookStruct

The WMMarkerHookStruct structure has the following members:

WMMarkerHookStruct Structure Members

Member	Description
`Int32 usage`	0= normal draw, 1= legend draw (others reserved).
`Int32 marker`	Marker number minus start (i.e., starts from zero).
`float x,y`	Location of desired center of marker
`float size`	Half width/height of marker
`Int32 opaque`	1 if marker should be opaque
`float penThick`	Stroke width
`STRUCT RGBColor mrkRGB`	Fill color
`STRUCT RGBColor eraseRGB`	Background color
`STRUCT RGBColor penRG`	Stroke color
`WAVE ywave`	Trace's y wave
`double ywIndex`	Point number; ywave[wyIndex] is the y value where the marker is being drawn.
`char winName[MAX_HostChildSpec+1]`	Full path to window or subwindow
`char traceName[MAX_OBJ_INST+1]`	Full name of trace or "" if no trace

When your marker function is called, the pen thickness and colors of the drawing environment of the target window are already set consistent with the penThick, mrkRGB, eraseRGB and penRGB members.

The winName and traceName members were added in Igor Pro 9.00 to provide access to trace userData. See **Trace User Data** on page IV-89.

Marker Hook Example

Here is an example that draws audiology symbols:

```
Function AudiologyMarkerProc(s)
   STRUCT WMMarkerHookStruct &s

   if( s.marker > 3 )
      return 0
   endif

   Variable size= s.size - s.penThick/2

   if( s.opaque )
      SetDrawEnv linethick=0,fillpat=-1
      DrawRect s.x-size,s.y-size,s.x+size,s.y+size
      SetDrawEnv linethick=s.penThick
```