In addition to creating local variables, a few operations, such as CurveFit and FuncFit, check for the existence of specific local variables to provide optional behavior. For example:

```
Function ExpFitWithMaxIterations(w, maxIterations)
   WAVE w
   Variable maxIterations

   Variable V_FitMaxIters = maxIterations

   CurveFit exp w
End
```

The CurveFit operation looks for a local variable named V_FitMaxIters, which sets the maximum number of iterations before the operation gives up.

The documentation for each operation lists the special variables that it creates or looks for.

Converting a String into a Reference Using $

The $ operator converts a string expression into an object reference. The referenced object is usually a wave but can also be a global numeric or global string variable, a window, a symbolic path or a function. This is a common and important technique.

We often use a *string* to pass the *name* of a wave to a procedure or to algorithmically generate the name of a wave. Then we use the $ operator to convert the string into a wave reference so that we can operate on the wave.

The following trivial example shows why we need to use the $ operator:

```
Function MakeAWave(str)
   String str

   Make $str
End
```

Executing

```
MakeAWave("wave0")
```

creates a wave named wave0.

Here we use $ to convert the contents of the string parameter str into a name. The function creates a wave whose name is stored in the str string parameter.

If we omitted the $ operator, we would have

```
Make str
```

This would create a wave named str, not a wave whos name is specified by the contents of str.

As shown in the following example, $ can create references to global numeric and string variables as well as to waves.

```
Function Test(vStr, sStr, wStr)
   String vStr, sStr, wStr

   NVAR v = $vStr       // v is local name for global numeric var
   v += 1
   SVAR s = $sStr       // s is local name for global string var
   s  += "Hello"
   WAVE w = $wStr       // w is local name for global wave
   w += 1
End

Variable/G gVar = 0; String/G gStr = ""; Make/O/N=5 gWave = p
Test("gVar", "gStr", "gWave")
```