

## Accessing Global Variables and Waves

Global numeric variables, global string variables and waves can be referenced from any function. A function can refer to a global that does not exist at compile-time. For the Igor compiler to know what type of global you are trying to reference, you need to declare references to globals.

Consider the following function:

```
Function BadExample()
    gStr1 = "The answer is:"
    gNum1 = 1.234
    wave0 = 0
End
```

The compiler can not compile this because it doesn't know what gStr1, gNum1 and wave0 are. We need to specify that they are references to a global string variable, a global numeric variable and a wave, respectively:

```
Function GoodExample1()
    SVAR gStr1 = root:gStr1
    NVAR gNum1 = root:gNum1
    WAVE wave0 = root:wave0

    gStr1 = "The answer is:"
    gNum1 = 1.234
    wave0 = 0
End
```

The SVAR statement specifies two important things for the compiler: first, that gStr1 is a global string variable; second, that gStr1 refers to a global string variable named gStr1 in the root data folder. Similarly, the NVAR statement identifies gNum1 and the WAVE statement identifies wave0. With this knowledge, the compiler can compile the function.

The technique illustrated here is called “runtime lookup of globals” because the compiler compiles code that associates the symbols gStr1, gNum1 and wave0 with specific global variables at runtime.

### Runtime Lookup of Globals

The syntax for runtime lookup of globals is:

```
NVAR <local name1>[= <path to var1>][, <loc name2>[= <path to var2>]]...
SVAR <local name1>[= <path to str1>][, <loc name2>[= <path to str2>]]...
WAVE <local name1>[= <path to wave1>][, <loc name2>[= <path to wave2>]]...
```

NVAR creates a reference to a global numeric variable.

SVAR creates a reference to a global string variable.

WAVE creates a reference to a wave.

At compile time, these statements identify the referenced objects. At runtime, the connection is made between the local name and the actual object. Consequently, the object must exist when these statements are executed.

<local name> is the name by which the global variable, string or wave is to be known within the user function. It does not need to be the same as the name of the global variable. The example function could be rewritten as follows:

```
Function GoodExample2()
    SVAR str1 = root:gStr1      // str1 is the local name.
    NVAR num1 = root:gNum1     // num1 is the local name.
    WAVE w = root:wave0        // w is the local name.

    str1 = "The answer is:"
```