

Dependency Assignment Statements

You can set up global variables and waves so that they automatically recalculate their contents when other global objects change. See Chapter IV-9, **Dependencies**, for details.

Operation Commands

An operation is a built-in or external routine that performs an action but, unlike a function, does not directly return a value. Here are some examples:

```
Make/N=512 wave1
Display wave1
Smooth 5, wave1
```

Operation commands perform the majority of the work in Igor and are automatically generated and executed as you work with Igor using dialogs.

You can use these dialogs to experiment with operations of interest to you. As you click in a dialog, Igor composes a command. This provides a handy way for you to check the syntax of the operation or to generate a command for use in a user-defined procedure. See Chapter V-1, **Igor Reference**, for a complete list of all built-in operations. Another way to learn their syntax is to use the Igor Help Browser's Command Help tab.

The syntax of operation commands is highly variable but in general consists of the operation name, followed by a list of flags (e.g., /N=512), followed by a parameter list. The operation name specifies the main action of the operation and determines the syntax of the rest of the command. The list of flags specifies variations on the default behavior of the operation. If the default behavior of the operation is satisfactory then no flags are required. The parameter list identifies the objects on which the operation is to operate. Some commands take no parameters. For example, in the command:

```
Make/D/N=512 wave1, wave2, wave3
```

the operation name is "Make". The list of flags is "/D/N=512". The parameter list is "wave1, wave2, wave3".

You can use numeric expressions in the parameter list of an operation where Igor expects a numeric parameter, but in an operation flag you need to parenthesize the expression. For example:

```
Variable val = 1.0
Make/N=(val) wave0, wave1
Make/N=(numpnts(wave0)) wave2
```

The most common types of parameters are literal numbers or numeric expressions, literal strings or string expressions, names, and waves. In the example above, wave1 is a name parameter when passed to the Make operation. It is a wave parameter when passed to the Display and Smooth operations. A name parameter can refer to a wave that may or may not already exist whereas a wave parameter must refer to an existing wave.

See **Parameter Lists** on page IV-11 for general information that applies to all commands.

User-Defined Procedure Commands

User-defined procedure commands start with a procedure name and take a list of parameters in parentheses. Here are a few examples:

```
MyFunction1(5.6, wave0, "igneous")
```

Macro and Function Parameters

You can invoke macros, but not functions, with one or more of the input parameters missing. When you do this, Igor displays a dialog to allow you to enter the missing parameters.