The Procedure Subtype

You can identify procedures designed for specific purposes by using a subtype. Here is an example:

```
Function ButtonProc(ctrlName) : ButtonControl
    String ctrlName

    Beep
End
```

Here, " : ButtonControl" identifies a function intended to be called when a user-defined button control is clicked. Because of the subtype, this function is added to the menu of procedures that appears in the Button Control dialog. When Igor automatically generates a procedure it generates the appropriate subtype. See **Procedure Subtypes** on page IV-204 for details.

The Parameter List and Parameter Declarations

The parameter list specifies the name for each input parameter. There is no limit on the number of parameters.

All parameters must be declared immediately after the function declaration. In Igor Pro 7 or later you can use inline parameters, described below.

The parameter declaration declares the type of each parameter using one of these keywords:

Variable	Numeric parameter
Variable/C	Complex numeric parameter
String	String parameter
Wave	Wave reference parameter
Wave/C	Complex wave reference parameter
Wave/T	Text wave reference parameter
DFREF	Data folder reference parameter
FUNCREF	Function reference parameter
STRUCT	Structure reference parameter
int	Signed integer parameter- requires Igor7 or later
int64	Signed 64-bit integer parameter - requires Igor7 or later
uint64	Unsigned 32-bit integer parameter - requires Igor7 or later
double	Numeric parameter - requires Igor7 or later
complex	Complex numeric parameter - requires Igor7 or later

int is 32 bits in IGOR32 and 64 bits in IGOR64.

double is a synonym for Variable and complex is a synonym for Variable/C.

WAVE/C tells the Igor compiler that the referenced wave is complex.

WAVE/T tells the Igor compiler that the referenced wave is text.

Variable and string parameters are usually passed to a subroutine *by value* but can also be passed by reference. For an explanation of these terms, see **How Parameters Work** on page IV-58.