If *flag* is 0, GetRTError returns an error code if an error has occurred or 0 if no error has occurred.

If *flag* is 1, GetRTError returns an error code if an error has occurred or 0 if no error has occurred and it clears the error state of Igor's runtime execution environment. Use this if you want to detect and handle runtime errors yourself.

If *flag* is 2, GetRTError returns the state of Igor's internal abort flag but does not clear it.

For *flag*=0 and *flag*=1, you can call **GetErrMessage** to obtain the error message associated with the returned error code, if any.

In Igor Pro 7.00 or later, using GetRTError(1) on the same line as a command that causes an error overrides the debugger "debug on error" setting and prevents the debugger from activating for that error.

Example

```
// Detect and handle a runtime error rather than allowing it to cause
// Igor to abort execution or invoke the debugger. The error must be
// recorded, using GetErrMessage, and cleared, using GetRTError,
// on the same line as the error.
Function Demo()
    String msg
    Variable err
    Make/O/N=(-2) wave0; msg=GetErrMessage(GetRTError(0),3); err=GetRTError(1)
    if (err != 0)
        Print "Error in Demo: " + msg
        Print "Continuing execution"
    endif
    // Do more things here
End
```

See also

The **GetErrMessage** and **GetRTErrMessage** functions.

GetRTErrMessage

`GetRTErrMessage()`

In a user function, GetRTErrMessage returns a string containing the name of the operation that caused the error, a semicolon, and an error message explaining the cause of the error. This is the same information that appears in the alert dialog displayed. If no error has occurred, the string will be of zero length. GetRTErrMessage must be called before the error is cleared by calling GetRTError with a nonzero argument.

For an overview of error handling, see **Flow Control for Aborts** on page IV-48.

See also

Flow Control for Aborts on page IV-48, **GetRTError**, **GetErrMessage**

GetRTLocation

`GetRTLocation(`*`sleepMS`*`)`

GetRTLocation is used for profiling Igor procedures.

You will typically not call GetRTLocation directly but instead will use it through FunctionProfiling.ipf which you can access using this include statement:

```
#include <FunctionProfiling>
```

GetRTLocation is called from an Igor preemptive thread to monitor the main thread. It returns a code that identifies the current location in the procedure files corresponding to the procedure line that is executing in the main thread.

Parameters

sleepMs is the number of milliseconds to sleep the preemptive thread after fetching a value. *sleepMs* must be between 0.001 and 100.

Details

The result from GetRTLocation is passed to **GetRTLocInfo** to determine the location in the procedures. This samples the main thread only and the location becomes meaningless after any procedure editing.