## XLLoadWave Output Variables

XLLoadWave sets the standard Igor file-loader output variables, V_flag, S_path, S_fileName, and S_waveNames. In addition it sets S_worksheetName to the name of the loaded worksheet within the workbook file.

## Excel Date/Time Versus Igor Date/Time

Excel stores date/time information in units of days since January 1, 1900 or January 1, 1904. 1900 is the default on Windows and 1904 is the default on Macintosh. Igor stores dates in units of seconds since January 1, 1904.

If you use the Treat all columns as date, Deduce from row, or Use column type string methods for determining the column type, XLLoadWave automatically converts from the Excel format into the Igor format. If you use the Treat all columns as numeric method, you need to manually convert from Excel to Igor format.

If the Excel file uses 1904 as the base year, the conversion is:

```
wave *= 24*3600                // Convert days to seconds
```

If the Excel file uses 1900 as the base year, the conversion is:

```
wave *= 24*3600                // Convert days to seconds
wave -= 24*3600*365.5*4        // Account for four year difference
```

The use of 365.5 here instead of 365 accounts for a leap year plus the fact that the Microsoft 1900 date system represents 1/1/1900 as day 1, not as day 0.

When displaying time data, you may see a one second discrepancy between what Excel displays and what Igor displays in a table. For example, Excel may show "9:00:30" while Igor shows "9:00:29". The reason for this is that the Excel data is just short of the nominal time. In this example, the Excel cell contains a value that corresponds to, "9:00:30" minus a millisecond. When Excel displays times, it rounds. When Igor displays times, it truncates. If this bothers you, you can round the data in the Igor wave:

```
wave = round(wave)
```

In doing this rounding, you eliminate any fractional seconds in the data. That is why XLLoadWave does not automatically do the rounding.

## Loading Excel Data Into a 2D Wave

XLLoadWave creates 1D waves. Here is an Igor function that converts the 1D waves into a 2D wave.

```
Function LoadExcelNumericDataAsMatrix(pathName, fileName, worksheetName,
                        startCell, endCell)
   String pathName         // Name of Igor symbolic path or "" to get dialog
   String fileName         // Name of file to load or "" to get dialog
   String worksheetName
   String startCell        // e.g., "B1"
   String endCell          // e.g., "J100"

   if ((strlen(pathName)==0) || (strlen(fileName)==0))
      // Display dialog looking for file.
      Variable refNum
      String filters = "Excel Files (*.xls,*.xlsx,*.xlsm):.xls,.xlsx,.xlsm;"
      filters += "All Files:.*;"
      Open/D/R/P=$pathName /F=filters refNum as fileName
      fileName = S_fileName         // S_fileName is set by Open/D
      if (strlen(fileName) == 0)    // User cancelled?
         return -2
      endif
   endif
```