```
      return -1
   endif

   Variable groupID              // HDF5 group ID will be stored here
   HDF5OpenGroup /Z fileID, groupPath, groupID
   if (V_flag != 0)
      Print "HDF5OpenGroup failed"
      HDF5CloseFile fileID
      return -1
   endif

   HDF5LoadData /O /A=attributeName /TYPE=(objectType) /N=tempAttributeWave /Q
                        /Z groupID, objectName
   result = V_flag               // 0 if OK or non-zero error code

   if (result == 0)
      Wave/T tempAttributeWave
      if (WaveType(tempAttributeWave) != 0)
         attributeValue = ""     // Attribute is numeric, not string
         result = -1
      else
         attributeValue = tempAttributeWave[0]
      endif
      KillWaves/Z tempAttributeWave
   endif

   // Close the HDF5 group
   HDF5CloseGroup groupID

   // Close the HDF5 file
   HDF5CloseFile fileID

   return result
End
```

Loading All Attributes of an HDF5 Group or Dataset

This function illustrates loading all of the attributes of a given group or dataset. The attributes are loaded into waves in the current data folder.

```
Function LoadHDF5Attributes(pathName, filePath, groupPath, objectName,
                        objectType, verbose)
   String pathName             // Symbolic path name - or ""
   String filePath             // File name, relative path or full path
   String groupPath            // Path to group, such as "/", "/metadata_group"
   String objectName           // Name of object whose attributes you want or "."
                        for the group specified by groupPath
   Variable objectType         // The type of object referenced by objectPath:
                               // 1=group, 2=dataset
   Variable verbose            // Bit 0: Print errors; Bit 1: Print warnings;
                               // Bit 2: Print routine info

   Variable printErrors = verbose & 1
   Variable printWarnings = verbose & 2
   Variable printRoutine = verbose & 4

   Variable result = 0         // 0 means no error

   // Open the HDF5 file
   Variable fileID             // HDF5 file ID will be stored here
   HDF5OpenFile /P=$pathName /R /Z fileID as filePath
```

```
if (V_flag != 0)
   if (printErrors)
      Print "HDF5OpenFile failed"
   endif
   return -1
endif

Variable groupID           // HDF5 group ID will be stored here
HDF5OpenGroup /Z fileID, groupPath, groupID
if (V_flag != 0)
   if (printErrors)
      Print "HDF5OpenGroup failed"
   endif
   HDF5CloseFile fileID
   return -1
endif

HDF5ListAttributes /TYPE=(objectType) groupID, objectName
if (V_Flag != 0)
   if (printErrors)
      Print "HDF5ListAttributes failed"
   endif
   HDF5CloseGroup groupID
   HDF5CloseFile fileID
   return -1
endif

Variable numAttributes = ItemsInList(S_HDF5ListAttributes)
Variable i
for(i=0; i<numAttributes; i+=1)
   String attributeNameStr = StringFromList(i, S_HDF5ListAttributes)

   STRUCT HDF5DataInfo di
   InitHDF5DataInfo(di)    // Initialize structure
   HDF5AttributeInfo(groupID,objectName,objectType,attributeNameStr,0,di)
   Variable doLoad = 0
   switch(di.datatype_class)
      case H5T_INTEGER:
      case H5T_FLOAT:
      case H5T_TIME:       // Not yet tested
      case H5T_STRING:
      case H5T_BITFIELD:   // Not yet tested
      case H5T_OPAQUE:     // Not yet tested
      case H5T_REFERENCE:
      case H5T_ENUM:       // Not yet tested
      case H5T_ARRAY:      // Not yet tested
         doLoad = 1
         break
      case H5T_COMPOUND:   // HDF5LoadData can not load a compound attribute
         doLoad = 0
         break
   endswitch
   if (!doLoad)
      if (printWarnings)
         Printf "Not loading attribute %s - class %s not supported\r",
                     attributeNameStr, di.datatype_class_str
      endif
      continue
   endif

   HDF5LoadData /O /A=attributeNameStr /TYPE=(objectType) /Q /Z groupID,
```