Subject: Re: reading wav files
Posted by davidf on Tue, 16 Jan 2001 02:11:12 GMT

View Forum Message <> Reply to Message

jnettle1 (jnettle1@utk.edu) writes:

> Does anyone have a routine/set of routines to read *.wav files in IDL? And if
> so, do you feel like sharing it?? :)

I'd try the READ_WAV routine built into IDL since
at least IDL 5.3.1. :-)

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: reading wav files
Posted by hahn on Tue, 16 Jan 2001 10:29:24 GMT

View Forum Message <> Reply to Message

jnettle1@utk.edu (jnettle1) wrote:

> Does anyone have a routine/set of routines to read *.wav files in IDL? And if
> so, do you feel like sharing it?? :)

I process wav files randomly, so I have some read/write routines
written on my own. However, these routines work on stereo encoded
16 bits per sample files only and ignore anything but the data chunk.
I can send you the routines by mail if needed.

The following structe defines the header. It is 44 bytes long. It
assumes that the data chunk follows the format descriptor block.

```
WAV_HDR = { WAVE_HEADER,   $
      ID1:'****',   LoF:0L,    ID2:'********',    LoH:0L, $
      FMT:0,       NChan:0,   SpSc:0L,        DpSc:0L,  $
      BypSa:0,     BipSa:0,   ID3:'****',      LoD:0L   }

; ID1     the letters "RIFF"
; LOF     bytes from here (i.e. from byte 8) to end of file
```

```
; ID2      the letters "WAVE" followed by "fmt "
;          "fmt " is the beginning of the format descriptor block
; LoH      Length of Header (here 16 Bytes format descriptor) that follows
;          ---- start of format header ----
; FMT      format tag, 1 = PCM, 257 = IBM mu Law, 258 = IBM a Law,
;                       259 = IBM Adaptive PCM (adpcm)
; NChan    channels (1 = Mono, 2 = Stereo ...)
; SpSc     samples per second
; DpSc     data bytes per second = samples per second * bytes per sample
; BypSa    bytes per sample = channels * (bits per sample / 8)
; BipSa    bits per sample
;          ---- end of format header ----
;          Now watch for "data": Other blocks will be skipped.
; ID3      the letters "data". This is the beginning of the data block
; LoD      data length, i.e. the number of bytes of the actual data.
;          ---- start of sampled data ----
;          for multi-channel samples 1st byte is channel 1, 2nd byte is
;          channel 2, 3rd byte channel 3 (or if 2 channels, channel 1 again)
;          Amplitude (y-values) from -127 to 127 or -32767 to 32767.
;          ---- end of sampled data ----
;          Comments may follow. Don't play 'em
;
```

Norbert