SoundHeader structure

#include <Sound.h>

typedef struct **SoundHeader** {

		Size	Offset	Description
Ptr	samplePtr;	4	0	if NULL then samples are in sampleArea
unsigned long	length;	4	4	length of sound in bytes
Fixed	sampleRate;	4	8	sample rate for this sound
unsigned long	loopStart;	4	12	start of looping portion
unsigned long	loopEnd;	4	16	end of looping portion
unsigned char	encode;	1	20	header encoding
unsigned char	baseFrequency;	1	21	baseFrequency value
char	sampleArea[1];	1	22	
} SoundHeader		24		

typedef SoundHeader ***SoundHeaderPtr**;

Field descriptions

samplePtr — A pointer to the sampled sound data. If the sampled sound is located in memory immediately after the *baseFrequency* field, then this field should be set to NULL. Otherwise, this field is a pointer to the memory location of the sampled sound data.

length — The number of bytes in the sampled sound data.

sampleRate — The rate at which the sample was originally recorded. The approximate sample rates are shown in theTable below. Note that the sample rate is declared as a Fixed data type, but the most significant bit is not treated as a sign bit; instead, that bit is interpreted as having the value 32,768.

loopStart — The starting point of the portion of the sampled sound header that is to be used by the **Sound Manager** when determining the duration of freqDurationCmd. These loop points specify the byte numbers in the sampled data to be used as the beginning and end points to cycle through when playing the sound.

loopEnd — The end point of the portion of the sampled sound header that is to be used by the **Sound Manager** when determining the duration of freqDurationCmd. If no looping is desired, set both loopStart and loopEnd to 0.

encode — The method of encoding used to generate the sampled sound data. The current encoding option values are

stdSH	standard sound header
extSH	extended sound header
cmpSH	compressed sound header

For a standard sound header, you should specify the constant stdSH. Encode option values in the ranges 0 through 63 and 128 to 255 are reserved for use by Apple. You are free to use numbers in the range 64 through 127 for your own encode options.

baseFrequency The pitch at which the original sample was taken. This value must be in the range 1 through 127. The Table in the section **Playing Frequencies** lists the possible baseFrequency values. The *baseFrequency* value allows the **Sound Manager** to calculate the proper playback rate of the sample when an application uses the freqDurationCmd command. Applications should not alter the *baseFrequency* field of a sampled sound; to play the sample at different pitches, use freqDurationCmd or freqCmd.

sampleArea An array of bytes, each of which contains a value similar to the values in a wave-table description. These values are interpreted as offset values, where 0x80 represents an amplitude of 0. The value 0x00 is the most negative amplitude and 0xFF is the largest positive amplitude. The samples are numbered 1 through the value in the length parameter.

The **Sound Manager** can play sounds sampled at any rate up to 64 kHz. The Table below lists approximate values for the most common sample rates. When you specify a value in the sampleRate field of a sound header, you should use the values in the third column.

SampleRate

Rate (kHz)	**Rate (Hz)**	**value (Fixed)**
5 kHz	5563.6363	0x15BBA2E8
7 kHz	7418.1818	0x1CFA2E8B
11 kHz	11127.2727	0x2B7745D1
22 kHz	22254.5454	0x56EE8BA3
44 kHz	44100.0000	0xAC440000