SOX

COLLABORATORS			
	TITLE : SOX		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY		January 8, 2025	

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

SOX

1.1 SOX(1)

Welcome to Amiga SOX, the Sound Exchange.

```
NAME
SYNOPSIS
DESCRIPTION
OPTIONS
FILE TYPES
EFFECTS
BUGS
NOTICES
AMIGA INFO
```

1.2 name

```
NAME
     sox - SOund eXchange - universal sound sample translator
```

1.3 synopsis

```
SYNOPSIS
     sox infile outfile
     sox infile outfile [  effect  [ effect options ... ] ]
     sox infile -e  effect  [ effect options ... ]
     sox [ general options  ] [ format options  ] ifile [  format
         options  ] ofile [ effect [ effect options ... ] ]

     General options:   [ -V ]   [ -v volume ]
     Format options:    [ -t filetype ]   [ -r rate ]   [ -s/-u/-U/-A ]
                        [ -b/-w/-l/-f/-d/-D ]   [ -c channels ]   [ -x ]
     Effects:
           copy
           rate
           avg
```

```
             stat
             echo   delay volume [ delay volume ... ]
             vibro   speed [ depth ]
             lowp   center
             band   [ -n ] center [ width ]
             highp   center
             reverse
```

1.4 description

```
DESCRIPTION
     Sox translates sound files from one format to another,  pos-
     sibly doing a sound effect.
```

1.5 options

```
OPTIONS
     The option syntax is a little grotty, but in essence:
          sox file.au file.voc
     translates a sound sample in SUN Sparc  .AU  format  into  a
     SoundBlaster .VOC file, while
          sox -v 0.5 file.au -rate 12000 file.voc rate
     does the same format translation but also lowers the  ampli-
     tude by 1/2 and changes the sampling rate from 8000 hertz to
     12000 hertz via the rate sound effect loop.

     File type options:

      -t  <string>          File type specification
      -r  <integer>         Sampling rate specification
      -s/-u/-U/-A           Sample type (style) spec
      -b/-w/-l/-f/-d/-D     Sample size spec
      -x                    Word order flag
      -c  <integer>         Number of channels

     General options:

      -e                    Effect flag
      -v  <float>           Volume change
      -V                    Verbose processing notification flag

     The input and output files may be standard input and output.
     This  is specified by '-'.  The -t type option must be given
     in this case, else sox will not know the format of the given
     file.   The  -t,  -r,  -s/-u/-U/-A,  -b/-w/-l/-f/-d/-D and -x
     options refer to the input data when given before the  input
     file name.  After, they refer to the output data.

     If you don't give an output file name, sox  will  just  read
     the  input  file.   This is useful for validating structured
     file formats; the stat effect may also be used  via  the  -e
     option.
```

1.6 -t

```
-t filetype
        gives the type of the sound sample file.
```

1.7 -r

```
-r rate    Give sample rate in Hertz of file.
```

1.8 suua

```
-s/-u/-U/-A
        The sample data is signed linear (2's complement),
        unsigned  linear,  U-law (logarithmic),  or A-law
        (logarithmic).  U-law and A-law are the  U.S.  and
        international  standards for logarithmic telephone
        sound compression.
```

1.9 bwlfdd

```
-b/-w/-l/-f/-d/-D
        The sample data is in bytes, 16-bit words,  32-bit
        longwords, 32-bit floats, 64-bit double floats, or
        80-bit IEEE floats.  Floats and double floats  are
        in native machine format.
```

1.10 -x

```
-x         The sample data is in XINU  format;  that  is,  it
           comes  from a machine with the opposite word order
           than yours and must be swapped  according  to  the
           word-size  given  above.  Only  16-bit and 32-bit
           integer  data  may  be  swapped.   Machine-format
           floating-point  data is not portable. IEEE floats
           are a fixed, portable format. ???
```

1.11 -c

```
-c channels
        The number of sound channels  in  the  data  file.
        This  may be 1, 2, or 4; for mono, stereo, or quad
        sound data.
```

1.12   -e

```
-e          after the input file allows you to avoid giving an
            output file and just name an effect.  This is only
            useful with the stat effect.
```

1.13   -v

```
-v volume  Change amplitude (floating point); less  than  1.0
            decreases,  greater  than 1.0 increases.  Note: we
            perceive  volume  logarithmically,  not  linearly.
            Note: see the stat effect.
```

1.14   -_v

```
-V          Print a description of processing phases.  Useful
            for  figuring out exactly how sox is mangling your
            sound samples.
```

1.15   file types

```
FILE TYPES
      Sox needs to know the formats of the input and output files.
      File  formats which have headers are checked, if that header
      doesn't seem right, the program exits  with  an  appropriate
      message.  Currently, the raw (no header), IRCAM Sound Files,
      Sound Blaster, SPARC .AU (w/header), Mac HCOM, PC/DOS  .SOU,
      Sndtool,  and Sounder, NeXT .SND, Windows 3.1 RIFF/WAV, Tur-
      tle Beach .SMP, and Apple/SGI AIFF and 8SVX formats are sup-
      ported.

       .aiff      AIFF files used on Apple IIc/IIgs, SGI, and Mac.
       .au        SUN Microsystems AU files.
       .hcom      Macintosh HCOM files.
       .raw       Raw files (no header).
       .ub, .sb, .uw, .sw, .ul
                  Various shorthands for raw types.
       .sf        IRCAM Sound Files.
       .voc       Sound Blaster VOC files.
       .auto      ''Meta-type'' for auto-type-detect.
       .smp       Turtle Beach SampleVision files.
       .wav       Windows 3.1 .WAV RIFF files.
```

1.16   .aiff

.aiff   AIFF files used on Apple IIc/IIgs and SGI. Note:
          the AIFF format supports only one SSND chunk. It
          does not support multiple sound chunks, or the
          8SVX musical instrument description format. AIFF
          files are multimedia archives and and can have
          multiple audio and picture chunks. You may need a
          separate archiver to work with them.

1.17 .au

.au   SUN Microsystems AU files. There are apparently
          many types of .au files; DEC has invented its own
          with a different magic number and word order. The
          .au handler can read these files but will not
          write them. Some .au files have valid AU headers
          and some do not. The latter are probably original
          SUN u-law 8000 hz samples. These can be dealt
          with using the .ul format (see below).

1.18 .hcom

.hcom   Macintosh HCOM files. These are (apparently) Mac
          FSSD files with some variant of Huffman compres-
          sion. The Macintosh has wacky file formats and
          this format handler apparently doesn't handle all
          the ones it should. Mac users will need your
          usual arsenal of file converters to deal with an
          HCOM file under Unix or DOS.

1.19 .raw

.raw   Raw files (no header).
          The sample rate, size (byte, word, etc), and style
          (signed, unsigned, etc.) of the sample file must
          be given. The number of channels defaults to 1.

1.20 .ub

.ub, .sb, .uw, .sw, .ul
          These are several suffices which serve as a short-
          hand for raw files with a given size and style.
          Thus, ub, sb, uw, sw, and ul correspond to
          "unsigned byte", "signed byte", "unsigned word",
          "signed word", and "ulaw" (byte). The sample rate
          defaults to 8000 hz if not explicitly set, and the
          number of channels (as always) defaults to 1.
          There are lots of Sparc samples floating around in

u-law format with no header and fixed at a sample
rate of 8000 hz. (Certain sound management
software cheerfully ignores the headers.) Simi-
larly, most Mac sound files are in unsigned byte
format with a sample rate of 11025 or 22050 hz.

1.21 .sf

.sf   IRCAM Sound Files.
          SoundFiles are used by academic music software
          such as the CSound package, and the MixView sound
          sample editor.

1.22 .voc

.voc   Sound Blaster VOC files.
          VOC files are multi-part and contain silence
          parts, looping, and different sample rates for
          different chunks. On input, the silence parts are
          filled out, loops are rejected, and sample data
          with a new sample rate is rejected. Silence with
          a different sample rate is generated appropri-
          ately. On output, silence is not detected, nor
          are impossible sample rates.

1.23 .auto

.auto   This is a ``meta-type'': specifying this type for
          an input file triggers some code that tries to
          guess the real type by looking for magic words in
          the header.   If the type can't be guessed, the
          program exits with an error message.   The input
          must be a plain file, not a pipe. This type can't
          be used for output files.

1.24 .smp

.smp   Turtle Beach SampleVision files.
          SMP files are for use with the PC-DOS package Sam-
          pleVision by Turtle Beach Softworks. This package
          is for communication to several MIDI samplers. All
          sample rates are supported by the package,
          although not all are supported by the samplers
          themselves. Currently loop points are ignored.

1.25 .wav

```
.wav     Windows 3.1 .WAV RIFF files.
         These appear to be very similar to IFF files,  but
         not  the same. They are the native sound file for-
         mat of Windows 3.1.  Obviously, Windows 3.1 is  of
         such  incredible importance to the computer indus-
         try that it just had to have its  own  sound  file
         format.
```

1.26 effects

```
EFFECTS
     Only one effect from the palette may be applied to  a  sound
     sample.   To do multiple effects you'll need to run sox in a
     pipeline.

      copy                           Copy input to output (default).
      rate                           Resample.
      avg                            Mix 4 channels to 2, or 2 to 1.
      stat                           Statistical check of input.
      echo  [ delay volume ... ]     Add echoing to a sound sample.
      vibro  speed  [ depth ]        Add vibrato.
      lowp  center                   Apply a low-pass filter.
      band  [ -n ] center [ width ]  Apply a band-pass filter.
      highp  center                  Apply a high-pass filter.
      reverse                        Reverse a soundfile.

     Sox enforces certain effects.  If the two  files  have  dif-
     ferent  sampling  rates, the requested effect must be one of
     copy, or rate, If the two files have  different  numbers  of
     channels, the avg effect must be requested.
```

1.27 copy

```
copy                          Copy the  input  file  to  the
                              output  file.  This  is  the
                              default effect if  both  files
                              have  the  same sampling rate,
                              or the rates are "close".
```

1.28 rate

```
rate                          Translate input sampling  rate
                              to  output  sampling  rate via
                              linear  interpolation  to  the
                              Least  Common  Multiple of the
                              two sampling rates.  This  is
                              the  default effect if the two
```

files have different sampling
rates.   This   is fast but
noisy.

1.29 avg

avg   Mix 4- or 2-channel sound file
                                 into 2- or 1-channel file by
                                 averaging the samples for dif-
                                 ferent speakers.

1.30 stat

stat   Do a statistical check on the
                                 input file, and print results
                                 on the standard error file.
                                 stat may copy the   file
                                 untouched from input to out-
                                 put, if you select an output
                                 file. The "Volume Adjustment:"
                                 field in the statistics gives
                                 you the argument to the -v
                                 number which will make the
                                 sample as loud as possible.

1.31 echo

echo [delay volume ...]   Add echoing to a sound sample.
                                 Each delay/volume pair gives
                                 the delay in seconds and the
                                 volume (relative to 1.0) of
                                 that echo. If the volumes add
                                 up to more than 1.0, the sound
                                 will melt down instead of fad-
                                 ing away.

1.32 vibro

vibro speed [depth]   Add   the   world-famous Fender
                                 Vibro-Champ sound effect to a
                                 sound sample by using a sine
                                 wave   as   the volume knob.
                                 Speed gives the Hertz value of
                                 the wave. This must be under
                                 30. Depth gives the amount
                                 the volume is cut into by the
                                 sine wave, ranging 0.0 to 1.0
                                 and defaulting to 0.5.

1.33 lowp

```
lowp center                  Apply a low-pass filter.  The
                             frequency response drops loga-
                             rithmically with  center  fre-
                             quency  in  the  middle of the
                             drop.  The slope of the filter
                             is quite gentle.
```

1.34 band

```
band [ -n ] center [ width ] Apply a band-pass filter.  The
                             frequency response drops loga-
                             rithmically around the  center
                             frequency.   The  width  gives
                             the slope of  the  drop.   The
                             frequencies  at center + width
                             and center - width  will  be
                             half  of their original ampli-
                             tudes.  Band defaults  to  a
                             mode  oriented to pitched sig-
                             nals, i.e. voice, singing,  or
                             instrumental  music.   The  -n
                             (for noise) option  uses  the
                             alternate  mode for un-pitched
                             signals.   Band    introduces
                             noise  in  the  shape  of  the
                             filter, i.e.  peaking  at  the
                             center  frequency and settling
                             around it.
```

1.35 highp

```
highp center                 Apply a high-pass filter.  See
                             the lowp effect for more info.
```

1.36 reverse

```
reverse                      Reverse the sample sequence in
                             a soundfile, so that the sound
                             plays "backwards".
```

1.37 bugs

BUGS
      The syntax is horrific. It's very tempting to include a
      default system that allows an effect name as the program
      name and just pipes a sound sample from standard input to
      standard output, but the problem of inputting the sample
      rates makes this unworkable.

1.38   notices

NOTICES
      The echoplex effect is Copyright (C) 1989 by Jef Poskanzer.

      Permission to use, copy, modify, and distribute   this
      software and its documentation for any purpose and without
      fee is hereby granted, provided that the above copyright
      notice appear in all copies and that both that copyright
      notice and this permission notice appear in supporting docu-
      mentation.   This software is provided "as is" without
      express or implied warranty.

1.39   amiga info

AMIGA INFO
      As compiled under SAS/C v6, SOX appears to work identically
      to the Unix version. Supplied in this archive are versions
      of SOX for 68000, 68020, 68881, and 68020+68881. All four
      versions contain AmigaDOS version strings to help identify
      which one you're using.

      I received no notification of the release of patchlevels 10
      and 11, so this was all on my own; I discovered the release
      myself and have made it Amiga-compilable. It seems to work.
      Because changes to the source were slightly more complex this
      time, I've supplied full source code. This source should
      still compile fine on any other platform, since my changes
      were only to the Amiga Makefile, files used only by the Amiga
      Makefile, or were enclosed inside preprocessor conditionals
      to filter out Amiga stuff on non-Amigas.

      NOTE (see below): This .guide file is not totally accurate;
      except for these end comments, it is identical to the Amiga
      SOX 3.1 release. I am not very into SOX any more, so I'm
      not very into rewriting its documentation either. I will
      continue to port SOX, but I will likely drop the .guides
      in the future since it's not easy to convert from Unix-style
      manuals.

      The SOX source code is maintained by Lance Norskog, with
      contributions from many others. Amiga ports (pl5, pl6) and
      modifications (pl7-9, 11) were performed by David Champion.
      My e-mail address is dgc3@midway.uchicago.edu; please send

any comments, bug reports, etc. to this address. If they
are Amiga-only problems, I will handle it myself and supply
necessary patches or recompiled binaries. If they are
problems with SOX as a whole, I will forward the report(s)
to the maintenance group.

The part of the README file that says that the Amiga port
was "done against old versions of SOX" is not true. This
release of Amiga SOX was made directly from the
generic/Unix release 6, pl11 distribution.

Note that in order to use pipes with Amiga SOX, you must
have a true piping shell such as Csh or WShell. The native
PIPE: device is not okay. Matt Dillon's FIFO: will work,
however.

This AmigaGuide file is not intended to be nice. For the
most part, it's simply a HyperText-ization of the file
'sox.txt'. Also, it has not been updated to conform to
the pl11 man page, since doing so is a pain and I'm not
all that interested in this program anymore.

    CHANGES   since last version
    MANIFEST

1.40 changes

With release 3.1, I decided to compile versions for the
68020 and 68000 instead of the '030 and '000. The
performance ratio of an '020 to a '000 is better than the
performance ratio of the '030 to an '020, and not much less
than '030 to '000. Essentially, '030 users lose little by
using an '020 binary, but '020 users gain much by not using
the '000 version. This solution, then, maximizes the
benefit distribution.

Changes from patchlevel 8 to patchlevel 9 include:
- new high-pass filter
- modified WAVE handler to accept RIFF's optional properties
- miscellaneous fixes elsewhere

Changes from pl 9 to pl 11 include:
- Sun ADPCM compress u-law support
- Other stuff. I haven't really looked.

1.41 manifest

Contained in this archive are the following:

  bin/csh/*2*.sh   Amiga Csh shell scripts for specific
        conversions.
  bin/csh/soxaliases.sh Amiga Csh aliases for specific conversions.

```
bin/dos/*2*   AmigaShell scripts for specific
     conversions.
bin/sox      Sound Exchange, runs on any Amiga.
bin/sox020    Sound Exchange for 68020 machines.
bin/sox881    Sound Exchange for Amigas with 68881/68882
                      floating-point processors.
bin/sox020881   Sound Exchange for 68020/6888[12] Amigas.

dist/*       Standard multi-platform distribution plus
     Amiga fixes which are NOT yet in the
     distribution.

doc/CHEATS    Original Unix CHEATS list.
doc/README    Original Unix README.
doc/sox.guide   AmigaGuide documentation.
doc/soxscripts.doc  Information on the supplied shell scripts.
doc/sox.doc   ASCII documentation.
doc/st.doc    ASCII documentation for libst.
doc/TIPS    Original Unix TIPS help file.
doc/TODO    Original Unix TODO file.
```

Information specific to the Amiga version is contained at the end of
the AmigaGuide document 'sox.guide'.

--dgc, 28 Feb 1994