Sounder 3.0 beta release 2
Beta release notes
Aaron Wallace
September 27, 1991

Changes since beta 1 are in bold

The user docs and new control panel are not finished yet, so this document will hopefully do both. Novices may find this a bit "rough;" for this I apologise.

For this beta, some familiarity with Sounder 2.0 or SoundTool is expected. I've included the docs for the previous version of Sounder; the section on how to play sound using SOUNDER.EXE is still valid.

The DSOUND.WRI file describes the new, expanded API of DSOUND.DLL. Most of the info is right, but is a bit dated. Definitely give DSOUND.WRI a gander; you'll find that adding sounds to programs in V-Basic, WordBASIC, Excel, or whatever is now almost trivial. No need to worry about huge memory, parsing file headers, or whatnot. For example:

```
PlaySoundFile(filename, 0, 0, 0, 0, 0)
```

will load and play filename, autodetecting which type of file it is. It doesn't get much easier than that. There's also support for loading sounds w/o playing them, playing sound resources, and more.

Note: if you have already installed a program that uses "another" version of DSOUND.DLL, such as SoundTool or Wired for Sound, this version should still work with such programs. However, definitely keep a copy of the "other" .DLL in case this one breaks too much.

Installation:

1) Copy the file DSOUND.DLL into your system directory, i.e. c:\win3\system.

2) Create a directory if necessary and copy all other files into it. \SOUNDER does well for me. Make sure this directory is in your path.

Now, decide which output device you want to use. Sounder now supports the internal speaker, the Sound Source, and the Sound Blaster/Thunder Card. **The DSOUND Control Panel (SNDCNTRL.EXE) can be used to configure and switch these for most normal settings.**

Installing for the internal speaker:

1) **Note: The internal speaker can be set up just by running the control panel. In particular, the delay value may have to be adjusted. DSOUND.WRI documents the WIN.INI settings if you prefer to go the manual route.**

Installing for the Sound Source:

1) Edit AUTOEXEC.BAT to load LPTREDIR.COM before Windows. LPTREDIR takes one parameter: the LPT port that the Sound Source is connected to:

```
...
LPTREDIR LPT1:
...
WIN
```

Reboot. If LPTREDIR complains that the LPT port is already being redirected, you're ok. If it says that there's no hardware for this port, you *may* be okay. LPTREDIR can be loaded only once.

2) **Run the DSOUND Control Panel and choose Sound Source. It will complain if LPTREDIR is not loaded. Set the Volume to 10 and adjust the Sound Source's volume knob for best fidelity. The delay setting has no effect on the Sound Source code. In case you cannot use LPTREDIR because of an existing network redirector, perform a manual setup:**
Create or find the [DSOUND] section of your WIN.INI file. Set the Volume= setting to 10, as the Sound Source's volume knob should be used instead. Make sure SoundDev=, if present, is 1. The Sounder 2.xx xDelayValue settings are not used. Now, for the fun part: set the Port= to the base I/O address of the LPT port, in decimal. The Hex value can be converted to decimal using the Windows calculator. For most computers, 0378h is LPT1, so you'd use 888 as the Port= value. Save WIN.INI and restart Windows.

Note: The Sound Source plays sounds at 7 KHz, so some sounds sampled at higher rates may sound "computerized." Problems arise when converting from 7 KHz to 11 KHz on the fly. 22 KHz sounds sound better.

Installing for the Sound Blaster:

1) **Run the DSOUND Control Panel and choose Sound Blaster. Set the Volume to 10 and adjust your amplifier or sound card for best fidelity. If your 'Blaster is not set for port 220h, you'll need to do a manual configuration:**
Create or find the [DSOUND] section of your WIN.INI file. Set the Volume= setting to 10, as the amplifier's volume knob should be used instead. Make sure SoundDev=, if present, is 2. The Sounder 2.xx xDelayValue settings are not used. Now, for the fun part: set the Port= to the base I/O address of the Sound Blaster, in decimal. The Hex value can be converted to decimal using the Windows calculator. Save WIN.INI and restart Windows.

Turning on Interrupts:

Normally, Sounder runs with interrupts off, which makes the sound sound great, but also kills background downloads. To leave interrupts on and kill sound quality (for the internal speaker, especially), add NoInts=0 to your WIN.INI file in [DSOUND]. NoInts=1 turns them back off.

Key Checking:

It is possible to allow the ESC key to abort sound playback. To enable this feature, add KeyCheck=8000 to WIN.INI in [DSOUND]. ESCKey should normally be 1, but can be the scan code of any key you'd like. KeyCheck=0 disables this feature. The actual KeyCheck value is the frequency of checking; if too high, there will be audible distortion (esp. on the internal speaker). Note that some programs "string" sounds together; if ESC is pressed and held, the key will be noticed by all subsequent sounds. If you just tap it, later sounds will still play. This will be fixable through an API change.

Fixes since release 1:

Delay re-implemented; seems different setups are still somewhat different. At least now, the delay will be 'close' on most machines by default.

Bugs in the 'Blaster and Sound Source drivers fixed.

Upcoming:

A lot of people have asked for recording capabilities, and the next beta should have these. The Sound Blaster and joystick will be supported. (Yes, the joystick!)

Known bugs:

The Control Panel is still quirky. The release version will hopefully auto-configure the Sound Blaster port, switch between drivers more gracefully, and allow the port to be set through an option box. Also, Shift is still there, but isn't implemented. Use Timer is no longer supported. For the story Auto configure, see DSOUND.WRI.

The playback tempo isn't always right. It seems to vary a bit between systems. Slower machines are more suceptible to this effect. **To fix, set the delay value appropriately.**

The API is a bit dated. Get/SetDelay will be in the final version, but they'll do something a bit different. Also, most of the PlaySoundXXX functions will return an abort flag (int), but this isn't fully implemented yet.

PlaySoundFile sometimes cannot break a file into pieces and play it.

The sound quality is not as good as it was. A new algorithm is being used, pending info from RealSound as to what they have patented.

There is no BASIC header file. It was on my MiniSport a few days ago... For now, here's an example of what you'll need. I've been able to use PlaySoundFile and LoadSoundFile/PlayImage with both Visual Basic and WordBASIC. Note that each Declare should be one full line:

```
Declare Function PlaySoundFile% Lib "DSOUND.DLL" (ByVal SFile$, ByVal SType%,
ByVal           Freq%, ByVal Samp%, ByVal Vol%, ByVal Shift%)
Declare Function GetDSoundVersion% Lib "DSOUND.DLL" ()
Declare Sub PlaySound Lib "DSOUND.DLL" (Samples As Any, ByVal Size&, ByVal Freq
%, ByVal                Samp%, ByVal Vol%, ByVal Shift%)
```

SOUNDER.EXE does not play SoundTool .SND files. It will. PlaySoundFile supports this format; SOUNDER.EXE does not since it hasn't been rewritten to use PlaySoundFile yet.

Terms:

By reading this you have agreed to a nondisclosure... No, not really. Beta testers (those who give me feedback) don't have to register the shareware version when it's out. Nor do people who have registered the previous versions. Other than that, distribute in unmodified form freely and enjoy! **Someone told me that the alternate form of registration I had mentioned here might get me in trouble; I still stand behind it, but don't mention it...**