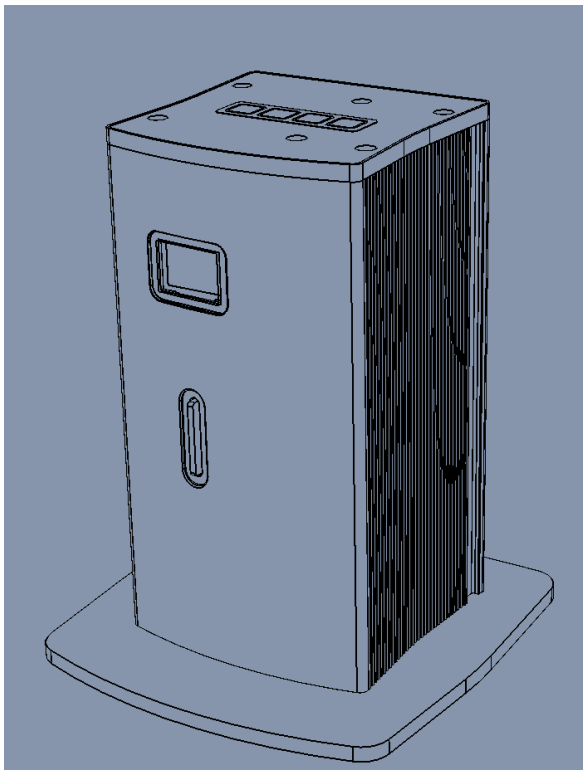
AudioFile User Guide

Version R03 December 2016



Cambridge Research Systems Limited

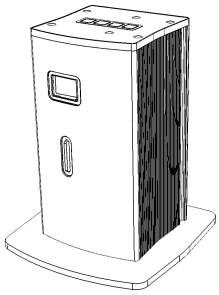
80 Riverside Estate
Sir Thomas Longley Road
Rochester
Kent
ME2 4BH
England

www.crsltd.com

Version	Date	Changed by:	Description of Changes
1.0	20/9/2012	E.Wallington	First issue
2.0	17/4/2014	J. Thomassen	Updated to current firmware version
3.0	8/12/2016	E. Wallington	Documented CDC interface in latest f/w.

Contents

Parts List



AudioFile – triggerable audio stimulus generator.

SDcard – holds settings and audio files.

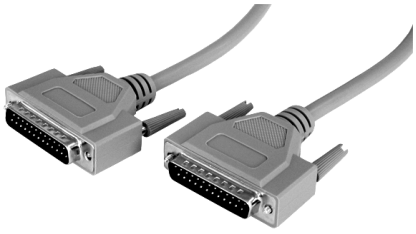
Put this in the SDcard socket on the front of AudioFile.

USB cable – connects from AudioFile to the host computer.

Instructions – printed version of these instructions.

Optional items

BNC trigger cable – connects from AudioFile to the visual stimulus generator system.



25 way track code cable – connects from AudioFile to the visual stimulus generator system.

Toslink digital audio (fibre optic cable) – connects from AudioFile to an external amplifier or amplified headphone system.

Analogue audio lead (various types) – connects from AudioFile to an external amplifier or amplified headphone system. The lead must finish in a stereo 3.5mm jack plug to fit AudioFile. The other end can be another jack plug (3.5mm or ¼ inch), or a pair of phono plugs, depending on the amplifier connections.

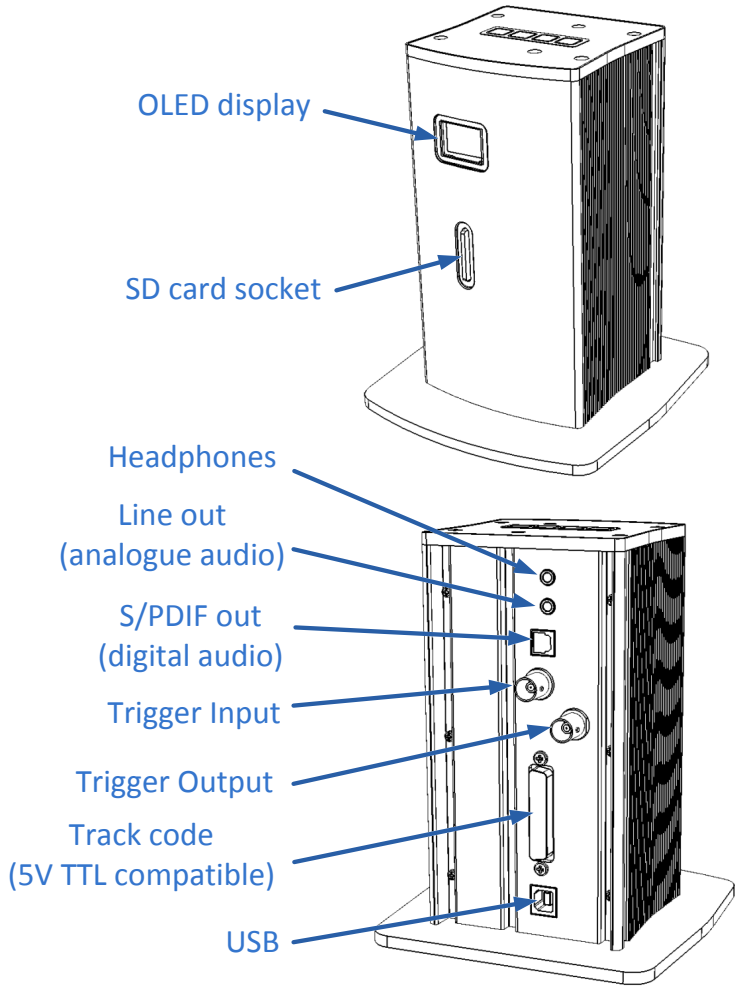
Introduction

Features

AudioFile is a zero latency triggerable audio stimulus generator (ASG).

- A colour OLED display shows status information.
- It has a slot for an SDcard for stimulus track storage. The files are stored as uncompressed 16 bit 44 kHz wav files to avoid introducing audio artefacts. A 2 GB card is supplied with AudioFile which gives over 3 hours of recorded stimulus.
- It is triggered with a TTL pulse, typically from a Visual Stimulus Generator (like the Cambridge Research Systems Visage or Bits#). A parallel digital code is also presented, which controls which stimulus track is to be played.
- Can also be controlled over a USB CDC interface.
- A trigger output indicates when a track is playing. This output can be offset to compensate for latency of external DACs.
- Audio output is via an optical SPDIF output, or conventional analogue line out.
- A separate headphone output is provided.
- USB bus powered, no separate PSU required.

Ports and connectors



Application

AudioFile can be used with the Cambridge Research Systems Visage or Bits# systems to provide a complete audio / visual stimulation solution. It is also capable of being used with any other visual stimulus generator that can create the necessary trigger pulses. Alternatively any stimulus software that can access a serial port can control AudioFile via a virtual serial port.

AudioFile is a self-contained system, it frees the experimenter from concerns about latency in generating the stimulus (which plague software only solutions).

When coupled with a MRI compatible audio delivery system it becomes part of a total MRI solution including the BOLDscreen range of MRI compatible displays.

Setting up the control Computer

Modes of operation

AudioFile has two modes of operation:

1) *AUDIOFile* mode. This is the default mode. In this mode the AudioFile plays audio tracks located on the SD card. Tracks can be selected, triggered and paused with practically no latency using the digital input port. In this mode the USB connection works as a mass storage device and can be used to load tracks onto the SD card. It can also be triggered via the USB connection over a CDC interface.

2) *USB audio* mode. In this mode the AudioFile can be used as a standard audio output device. The USB connection is used to feed the audio stream from the host computer to the AudioFile and does thus not expose the mass storage interface. Please note that when in this legacy mode, latency does depend on the host operating system and drivers. However, the trigger out BNC port will go high whenever a stream is playing to show when a track started and ended. A threshold can optionally be set in the config.xml file, to determine the start of the audio with greater precision (due to digital filtering the host stream often starts considerably before the start of the audio). Output values less than the threshold level will not be considered as the start of the stimulus and will not cause the trigger output to go high:

```
<Entry USBTriggerThreshold="0"/>
```

Configuring modes of operation

AudioFile is shipped with *AUDIOFile* mode active. This means that the USB interface shows AudioFile as a mass storage device. Start by plugging in the USB lead from the AudioFile to the host computer. The contents of the SD card will then appear as another volume called "*AUDIOFILE*" (AudioFile acts as a SD card reader in this mode).

If you want to change to *USB Audio* mode, edit the **config.xml** file located on the AudioFile volume. You can edit this file with any suitable text editor (for instance Notepad on Windows, or Xcode on OSX). Typical contents as follows:

```
<?xml version="1.0"?>
<AUDIOFILE_CFG>

      #DeviceType Is Either "AUDIOFile" Or "USB Audio"
      <Entry DeviceType="AUDIOFile"/>

      # These settings control which USB interfaces are enabled when in AudioFile mode.
      # The CDC interface (serial interface), allows control of the AudioFile without
      # use of the track select and trigger inputs.
      # This also allows access to trigger codes above 255, and dynamic reloading of different tracks.
      <Entry USB_MSDEnable="1"/>
      <Entry USB_CDCEnable="1"/>

      # This setting controls the volume of the analogue headphone output only
      <Entry Volume="75"/>

      # The following setting controls the threshold at which the trigger output
      # activates when in the "USB Audio" mode. If set to zero then the trigger activates
      # whenever a USB audio stream is detected, which may preceed the start of the played audio
      # (this depends on host computer sofware).
      # Setting has no effect in AUDIOFile mode.
      <Entry USBTriggerThreshold="0"/>

      # Delay of trigger output to compensate for latency in an external DAC. Units are 100us.
      # This would be set to a non zero value if the digital audio output is used.
      # Consult your external DAC documentation to obtain this setting.
      # Set this value to zero if the analogue outputs are used.
      <Entry USBTriggerDelay="0"/>

</AUDIOFILE_CFG>\
```

Edit the entry called "*DeviceType"* to read "*AUDIOFile*" or "*USB Audio*" as desired.

Once you have finished editing the **config.xml** file, exit the text editor (saving the file when prompted). Then re-boot the AudioFile by disconnecting the USB lead, waiting for 2 seconds, then reconnecting it.

NB. To leave this mode the SDcard needs to be removed and placed into a separate card reader (not included). Alternatively, a second SD card can be used for the *USB audio* mode.

Laptop computers

With a laptop computer, you may receive a message that there is insufficient power available. AudioFile requires 500mA to be available from the USB connection. In this case try (in this order):

1. A different USB socket on the laptop.
2. Connecting the laptop to its mains charger.
3. Or finally connect AudioFile by a powered USB hub.

Maintenance

Performing firmware updates

Firmware updates are easily performed by copying the firmware update file to the subdirectory called *Firmware* on the SD card. Once the copy has been completed, restart the AudioFile by disconnecting then reconnecting the USB lead. AudioFile will then update its firmware with the new file.

Setting up the audio stimulus

Playlist XML file

The file **Playlist.xml** contains the details of how the code on the parallel connector, maps to the file names of the wav files that are to be played. It is possible to have codes above 255 (up to 499), but these are only usable over the CDC interface.

You can edit the file with any text editor. Typical contents as follows:

```
<?xml version="1.0"?>
<AUDIOFILE_PLAYLIST>
   <PLAYLIST1>
      <Entry Folder="Audio Tracks"/>
      <Entry Code001="track1.wav"/>
      <Entry Code002="track2.wav"/>
      <Entry Code003="track3.wav"/>
      <Entry Code004="track4.wav"/>
      <Entry Code005="track5.wav"/>
   </PLAYLIST1>
   <SYSTEM>
       <Entry UseDigitalInputD0="FALSE"/>
       <Entry StopCode="0"/>
      <Entry SwapChannels="FALSE"/>
      <Entry SDRAMTest="FALSE"/>
   </SYSTEM>
</AUDIOFILE_PLAYLIST>
```

System entries:

- **UseDigitalInputD0** controls whether input bit 0 is used or not. For compatibility with Bits# and VSG Visage systems, D0 is normally disabled and bits D1 to D7 are the code (128 values). If **UseDigitalInputD0** is set true then 256 code values can be used.

- **StopCode** sets a special code which stops the current track from playing.
- **SwapChannels** reverses left and right channels.
- **SDRAMTest**, turns on or off a test of the internal RAM buffer. In normal operation this will be set to FALSE.

Loading the stimulus tracks

The user is free to create whatever directory structure necessary to hold the stimulus wav files. It is recommended that they are *not* located in the root of the AudioFile volume, create subdirectories for them.

Connecting the audio output

Connecting headphones

Any suitable pair of headphones can be plugged in to the headphone socket.

Connecting an external amplifier (for headphones or speakers)

An external amplifier can be connected to the digital or line level analogue outputs of AudioFile as appropriate.

Considerations for Phase shift (delay) of audio signals

The digital audio output of AudioFile has no significant phase shift compared to the trigger signal (< 100us). Subsequent conversion to analogue signals can however result in significant extra latency, depending on the technology used in the external digital to analogue converter (DAC).

Typically audio DAC's use sigma-delta type technology which by design introduces a delay to the audio signals. Professional audio equipment often has lower latency; alternatively if the external latency can be determined from the equipment specifications, then there is an xml setting in the config.xml file that can be used to delay the trigger output in increments of 100us to match the external DAC delay:

```
<Entry USBTriggerDelay="0"/>
```

Selecting and triggering the audio output

Audio tracks are selected by setting the binary state of the digital input pins of the parallel track code connector to match the track code number. After selecting the track, playback can be started by sending a trigger pulse (TTL) to either the "Trig In" BNC connector or trigger input on pin 9 of the parallel track code connector.

If using AudioFile with a Cambridge Research Systems "Bits# " or "Visage" visual stimulator it is recommended to use the parallel track code connector for selecting tracks as well as triggering the onset. The ports are laid out so that a regular straight-through parallel can be used for this purpose (DB 25 M/M).

If using any other equipment for selecting audio tracks, the appropriate connections must be made. Please refer to the pin layout shown in Table 1. For example, if used with a standard parallel port connections should be made as shown in Table 2.

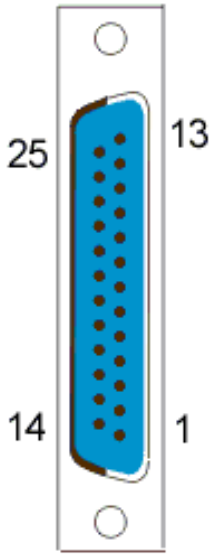
Note that when using the BNC connector for triggering make sure you don't have pin 9 of the track code connector connected to anything (and vice versa) as this might unintentionally hold the level of the input trigger and thus prevent the trigger pulse from being registered.

If nothing is connected to the track code connector AudioFile will automatically select the first track in the playlist. This is convenient if only one audio track is needed. The track can then easily be started by sending a trigger to the trigger input on the BNC connector.

Trigger Output

The “Trig Out” BNC connector as well as trigger out on pin 24 of the parallel track code connector provides TTL compatible outputs if you wish to monitor when audio is being played.

Table 1: Parallel track code connector pinout



Pin number	Function
1	Dout0
2	Dout1
3	Dout2
4	Dout3
5	Dout4
6	Dout5
7	Dout6
8	Dout7
9	Trigger In
10	0V
11	No connection
12	0V
13	0V
14	reserved
15	reserved
16	reserved
17	reserved
18	reserved
19	reserved
20	reserved
21	reserved
22	No connection
23	No connection
24	Trigger Out
25	reserved

Table 2: Parallel port to AudioFile connector cable:

DB25M (plug) pin#	DB25M (plug) pin#
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
12	18
13	19
18	12
19	13

Using the CDC interface

CDC commands

The CDC interface enables the AudioFile to be controlled without connecting the trigger input and track code connectors. This also enables a greater number (500) tracks to be played, and the tracks themselves to be changed during an experiment.

The CDC interface creates a virtual serial port between the computer and the AudioFile. Commands are simple text commands preceded with the $ symbol. The commands echo the command, with the current settings delimited by semicolons. The commands are not case sensitive.

If a command requires parameters then follow the command with an equals sign and enclose the parameter(s) in square brackets, thus:

$LoadTrack=[499,Audio Tracks\track499.wav]

Enabling the interface

Enable the CDC interface by editing the Config.xml file in the firmware subdirectory. On current operating systems this can be enabled simultaneously with the mass storage. On legacy operating systems you can only enable one interface at a time:

```
<Entry USB_MSDEnable="1"/>
<Entry USB_CDCEnable="1"/>
```

$Help

This command just lists the commands that are available. It does not take any parameters.

$Starttrack

This is probably the most useful command. You call this command with the desired trigger code, and the track corresponding to that code is played. The trigger output is active while the track is playing:

$starttrack=[499]

$Starttrack;499;

$Stoptrack

This command stops the current playing track. If no track is currently playing it does nothing. It is equivalent to asserting the stop-code on the track code connector.

$Stoptrack

$Stoptrack;

$Playlist

You call this command with the filename (including extension) of a new playlist file. Track entries in the new playlist will replace currently loaded tracks if the same trigger code is re-used. Trigger codes that are not present in the new playlist are not affected. You can therefore use the playlist to replace just a few tracks if required.

Please note that in order to achieve the zero latency performance, AudioFile pre-buffers the new audio tracks. This command can therefore take a significant amount of time to execute if lots of trigger codes are given. The actual time depends on the speed of SD card that is used. For instance, to pre-buffer the maximum 500 tracks from a class 4 card takes just under 2 minutes. The command does not give its return until the pre-buffering has completed and the experiment can resume.

Also note that changes to the mass storage volume will not take effect until the AudioFile is restarted. It is therefore not possible to copy files to the mass storage volume, then immediately load them . This is because reading the file allocation table cannot be done while the host computer has access. If you need to change or add to the actual files on the SDcard, once you have made the changes then eject the drive and issue the $Debug command to restart the Audiofile.

$LoadTrack

LoadTrack is similar to the playlist command, but it allows the file for one trigger code to be changed, without creating a playlist xml file.

Like the Playlist command this returns only after the new track has been pre-buffered. However as only one track is being pre-buffered the delay is shorter (typically a few hundred milli-seconds).

$ProductType

This will return “AudioFile” and enables the device to be identified when several different CRS devices are using CDC connections.

$FirmwareDate

This returns the date of the currently loaded firmware.

$USB_massStorage

This command is useful on legacy operating systems that cannot support the mass storage and CDC interfaces concurrently. When executed the CDC interface is closed and the mass storage volume appears instead.

$Debug

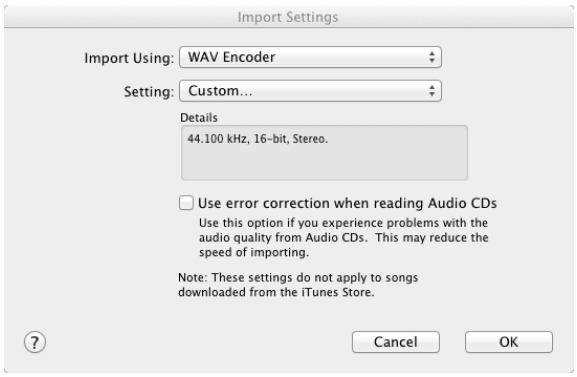
This command causes the AudioFile to restart. If the mass storage volume is enabled then eject the volume before executing this command to avoid generating surprise removal warnings from the operating system.

Generating the stimulus files

Using iTunes

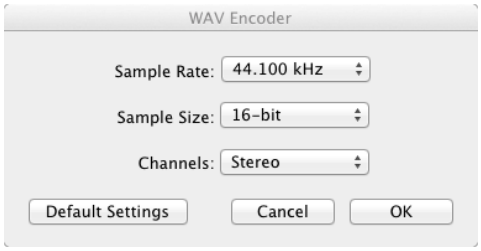
iTunes can create wav files from any unprotected file in the iTunes library. It can also create wav files from direct from an audio CD. By default iTunes uses the compressed AAC encoder. To enable the wav encoder instead you need to change the "Import Settings". These are located in the "General Preferences" (shown here under OSX 10.7)

Click the "Import Settings" button.



Then change the "Import Using" drop down to "WAV encoder".

Then go to the Setting drop down and select "Custom…"



In this window select 44.1kHz, 16-bit and Stereo.

Using MATLAB

MATLAB can read and write wav files using the *waveread* and *wavwrite* functions.

When writing files set the sample rate to 44100 Hz thus:

```
wavwrite(variable_name,44100,filename);
```

When reading files, make sure they are recorded with a sample rate of 44100Hz.

```
Variable_name = wavread(filename);
```

Safety

Life support applications

AudioFile should NOT be used in situations where failure of the device would constitute a hazard. It is designed for research applications only, and like any other regular electronic device could fail at any time, without warning.

Magnetic fields

AudioFile is MR unsafe. Do not take AudioFile or any of its leads into the magnet room of an MRI scanner.

Cleaning

Clean external components of LiveTrack with a damp cloth only. Do NOT allow fluids to enter any of the LiveTrack components. Do not sterilise in an autoclave.

Servicing

DO NOT ATTEMPT TO DISMANTLE AudioFile. It contains no user serviceable components, refer all servicing to Cambridge Research Systems.

Contact

Cambridge Research Systems Limited

Telephone: +44 1634 720707

Website: www.crsltd.com

80 Riverside,

Sir Thomas Longley Road,

Rochester,

Kent

United Kingdom

ME2 4BH