RABID MANTIS

Waveplayer

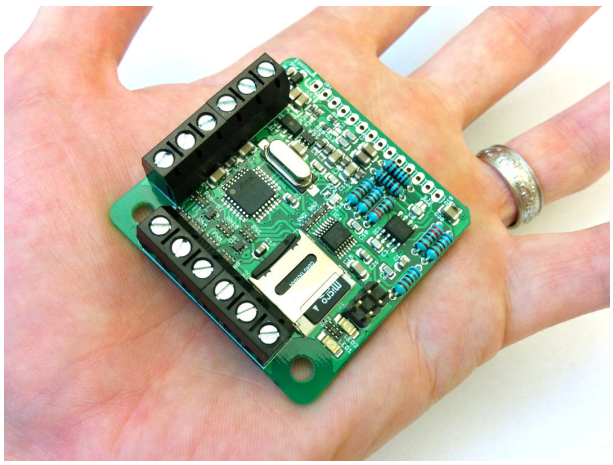
General-purpose standalone audio playback board.

Features

Measured SNR of >90dB at 3.9Vpp, 10Hz - 24kHz.
Very low THD (estimated <0.005% at 3.9Vpp out, 1kHz).
6 configurable input triggers on falling/rising flank.
Analog input for direct logarithmic volume control.
Wide range of sample rate, bit depth, and channel configurations supported.
Optional low-power mode reduces idle-state current draw to ≈ 1.7 mA.
Supports most microSD and microSDHC cards and sizes (up to 32GiB).
Header row allows for direct connection to Cheapamp board.

Description

Waveplayer is a general-purpose, standalone, stereo audio playback board. Wave files are read from a removable microSD or microSDHC card (up to 32GiB) with a FAT12/FAT16/FAT32 file system. Files with any combination of mono or stereo, 8-bit or 16-bit, and 8/11.025/16/22.05/32/44.1/48kHz sample rates are supported.
Waveplayer's analog output has an SNR level exceeding 90dB at full output amplitude and an estimated THD typically under 0.005% (3.9Vpp, 1kHz). A low-power mode allows for battery powered applications, with an idle-state current draw of typically 1.7 mA.
Six user-configurable inputs are available for triggering events (play, pause, next track, etc) on a rising or falling flank. One of these inputs can be used as an analog input for controlling the playback amplitude from −100dB to 0dB with a linear change in voltage.
A header row allows for directly connecting Waveplayer to a Cheapamp board (see `http://www.rabidmantis.se/cheapamp/`), for a complete audio amplifier system.
An integrated serial interface can be used for more complex control and allows for navigating the filesystem, controlling the playback state, and changing system settings.



Contents

Copyright Notice

Hardware Revision History

1.1.0 Initial released version.

Software Revision History

1.0.0 Initial version.

Documentation Revision History

1.0.0 Initial version, applies to hardware revision 1.1.0.

Absolute Maximum Ratings

PARAMETER	SYMBOL	RATING
Supply voltage	V_{in}	+26V[a] to -30V[b]
Control line/UART/Cheapamp control lines input voltage	$V_{inp,ctrl}$	+3.8V to -0.5V[c]
Analog output voltage DC bias	$V_{out,bias}$	±16V (ESD-protected[d])
Analog output short circuit current time	t_{SC}	Continuous

[a] At high (>20V) supply voltages the thermal dissipation of the primary voltage regulator IC3 may become problematic in small enclosures.
[b] Device is fully protected against reverse input polarity.
[c] Terminal block inputs are projected to withstand IEC 61000-4-2 L4 ESD discharges (IE. 15kV air discharge) and will withstand limited long-term over-voltage. Cheampamp control lines are not ESD protected.
[d] ESD protection level untested, likely to withstand minor ESD events.

Assembly

Waveplayer is relatively simple to assemble, with no particularly difficult packages. Most components have SMD carriers, though some through-hole components are used. The microprocessor requires programming before the board can be used; this can be done with any programmer capable of programming the AVR XMEGA series[1].

See the section Bill of materials for the components used and figure 7 on page 25 for their placement. All I/O terminal connections are described in the section Pin Description. Finally, the section Operation describes the internal operation of the device.

Warning: Be sure to follow the instructions below if using Waveplayer together with Cheapamp!

Waveplayer can be assembled in a few different ways depending on the desired functionality;

Mono mode Reduces the number of components mounted if only mono audio output is required. In this case, simply leave R10, R11, R12, R13, R14, R15, R16, R17, R18, C8, C9, C10, and C13 unmounted and short solder jumpers SJ2 and SJ1.

External 5V supply and no low-power mode If an external 5V supply is available and the device does not need to conserve power when playback is inactive (see Electrical Characteristics for current consumption levels) some components can be left unmounted. Leave F1, D1, C16, IC3, C14, C15, and IC5 unmounted and short solder jumper SJ4. Supply external power (4.5V – 5.5V) to pin 3 of JP1 (positive) and pin 2 (ground).

[1] Such as (among others) the AVRISP mkII, AVR Dragon, and JTAGICE3

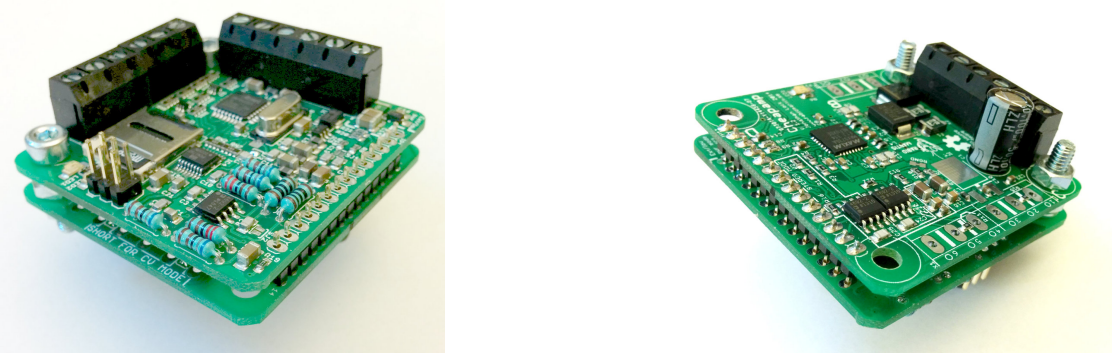


Figure 1: Waveplayer used together with Cheapamp. Note the use of a 14-pin 2.54mm header to electrically connect the boards and two M3 or M4 machine screws to mechanically hold the boards at a suitable spacing.

Standalone mode If Waveplayer is intended to be used stand-alone and the analog output ground (Pin 6 on JP1) is not externally connected to power ground (X3 pin 1) **SJ3 must be shorted!** This solder jumper locally connects the analog and power grounds, ensuring that digital signals to the DAC are correctly passed through, however, should some external device also connect power ground and analog ground this will potentially create a ground loop which may significantly reduce total performance. If SJ3 is shorted C18 can be left unmounted.

Cheapamp mode To use Waveplayer together with Cheapamp some steps must be undertaken; SJ3 should be left open (not shorted), Waveplayer's JP1 header row should be connected to Cheapamp's JP11 header row and oriented so that the chamfered corners of the PCB's match up (IE. the component sides of both PCB's shall be facing away from each other, see figure 1). **Furthermore, on Cheapamp (hardware revision 1.2) IC1, R5, and R7 MUST be left unmounted! Additionally, due to the relatively high output amplitude from Waveplayer, R2 and R4 can suitably be set to 3.3kΩ. Finally, X1 and X3 may be left unmounted as they are completely unused.** (Decreasing the internal gain in Cheapamp allows for an increased output amplitude from Waveplayer for the same total output volume, which will result in an improved SNR).

Analog resistors Either through-hole metal-film (R1, R3, R4, R5, R10, R12, R13, and R14) **OR** thin-film SMD (R2, R6, R7, R8, R11, R15, R16, R17) resistors can be used for the analog filter portion of Waveplayer. Typically thin-film SMD resistors are easier to mount and available at lower cost, however through-hole resistors are more commonly available. The performance of Waveplayer is virtually unaffected by the choice of resistor, leaving the choice up to personal preference. However, **be sure to mount only one type of resistor, not both!**

Programming the microprocessor

Depending on the platform and programmer used the actual commands to program the device vary. If using Atmel Studio be sure to upload the .hex file containing the software binary and set the device fuses as follows;

FUSEBYTE1 0x0A

FUSEBYTE2 0xFD

FUSEBYTE4 0xF1

FUSEBYTE5 0xDB

FUSEBYTE6 0xFF

Pin Description

Connections are as follows;

HEADER	PIN	DESCRIPTION
X1	1 - GND	Power ground connection.
	2 - IN0	General-purpose input 0 or analog volume control input.
	3 - IN1	General-purpose input 1
	4 - IN2	General-purpose input 2
	5 - IN3	General-purpose input 3
	6 - IN4	General-purpose input 4
X3	A - V_+_IN	Positive power voltage connection
	B - OUTR	Right audio output
	C - OUTL/MONO	Left audio output / mono audio output
	D - UART_RX	UART RX (receive) input
	E - UART_TX	UART TX (transmit) output
	F - IN5	General-purpose input 5
X2	-	microSD card connector
J1	-	Microprocessor PDI programming header
JP1	(1) V+ IN	Positive power voltage connection (connected to X3 pin A)
	(2) GND	Power ground connection (connected to X1 pin 1)
	(3) 5V SUP	+5V power output (active when low power mode disabled or playback active). Alternately, +5V power input when configured for external 5V supply (see Assembly).
	(4) $\overline{\text{SHDN}}$	Cheapamp shutdown output, indicates when the +5V power output is active.
	(5) $\overline{\text{MUTE}}$	Cheapamp mute output, indicates when audio playback is active.
	(6) AGND	Analog (audio) ground connection.
	(7) OUTL/MONO	Left/mono audio output.
	(8) OUTR	Right audio output.
	9+	[not connected]

Software Interface

Waveplayer uses the Rabidmantis CLI (command line interface) for controlling the device configuration and playback state. This interface is similar to a shell interface commonly found in computing settings in that commands can be executed with a variable number of arguments.

The text-based CLI interface can be driven either via a plain-text configuration file placed on the removable memory card (which allows for setting up the device to some configurable default settings) or via the UART interface (which allows for controlling the device in real-time; configuring device settings as well as navigating the file system, triggering playback, and so on). Regardless of the method of accessing the interface, identical syntax and commands are used.

The CLI allows for both executing commands as well as setting up device parameters; details about these are described more in the following subsections. The CLI indicates it is ready to accept a new command by printing the "`>`" character; commands can be entered and executed freely after this indicator. All commands sent to Waveplayer are line-based, and typically of the format[2] `command -argument1 value1 <newline>`, where <newline> is typically generated by pressing the enter key on the keyboard and is from here on in implied (not shown). Some commands may expect several arguments and values, while others need none. The order of arguments have no effect. Additionally, only the first appearance of an argument is used while unused arguments are ignored.

A list of all valid commands can be generated by calling the `help` command, while details for each individual command can be displayed by calling `help <command>`.

Waveplayer can be easily used with both character-based terminal applications (like PuTTY and Minicom) and line-based terminal applications (like CuteCom), as all valid input is directly echoed to the terminal and a rudimentary input-buffer is implemented, allowing the use of the backspace key to remove characters.

Valid commands

COMMAND	REQUIRED ARGUMENT	OPTIONAL ARGUMENT	DESCRIPTION
help	none	<command>	Lists all valid commands if no argument specified (eg. *'help'*). Lists information about a specific command if argument specified (eg. *'help get'*)
busy	none	none	Displays whether file playback is currently in progress or not.
stop	none	none	Stops any file currently being played. If no file is currently being played this has no effect.

[2] Waveplayer's CLI is completely text based and will only accept input characters in the ASCII range of 0x20 to 0x127 (0-9, a-z, A-Z, and all other printable ASCII characters such as !, ", and %), 0x08 (backspace), 0x0B (new line), and 0x0D (carriage return). All other inputs are regarded as invalid and will result in the output of a 0x07 character (system bell) to the host PC and are otherwise ignored.

COMMAND	REQUIRED ARGUMENT	OPTIONAL ARGUMENT	DESCRIPTION
pause	none	none	Pauses any file currently being played and stores the current playback location. If there is no file currently being played this has no effect. Note; **if low-power mode is enabled the device will NOT enter low-power mode when paused**!
resume	none	none	Resumes a paused file and continues playback from the paused location. If no file has been paused this has no effect. Note; if playback is interrupted with the stop command the file cannot be resumed from its previous location.
playpause	none	none	Pauses any file currently being played and resumes any file currently paused. If a file is neither being played back nor paused this has no effect.
get	none	none	Lists all device parameters and their settings (see Waveplayer settings).
set	<parameter name> <value>	sudo, min or max	Sets a device parameter setting to a certain value (eg. *'set -p param1 5'* would set a parameter with name param1 to a value of 5) (see Waveplayer settings). Optionally overrides the default maximum/minimum values (eg. *'set -sudo -max -p param1 500'* would override the maximum value and change it to 500). **Use minimum/maximum override with caution.**
ls	none	none	Lists all files, their file size, and directories in the current directory. (see example 2 on page 11).
cd	<directory>	none	Changes the current directory. '*cd foo*' will enter a directory *foo* in the current directory, '*cd ..*' will travel on directory up towards the root directory.
diskinfo	none	none	Lists various memory card information such as size, manufacturer, serial number, and so on.
play	<filename>	none	Starts playback of a given file. '*play bar.wav*' will start playback of a file *bar.wav* in the current directory.
next	none	none	Starts playback of the next file as controlled by the shuffle parameter. If configured for shuffled playback this will start playback of a random file in the current directory, otherwise this will start playback of the next file in ASCII-code order[3].

[3]This is similar to alphabetical order with the primarily the following deviations;

- All uppercase letters precede lower case letters (eg. '*Z.wav*' is ordered before '*a.wav*').
- Digits and many punctuation marks precede letters (eg. '*4.wav*' is ordered before '*one.wav*').
- Numbers are sorted as their string representation (eg. '*10.wav*' is ordered before '*2.wav*').

If playback order is important one simple method of maintaining order is to name all files '*001.wav*' through '*999.wav*'. If more than 999 files are needed add a suitable number of padding zeros, ie. start with '*0001.wav*' to support ordering of up to 9999 files.

Waveplayer settings

PARAMETER	MIN.	MAX.	DEFAULT	DESCRIPTION
verbose	0	1	1	If enabled (set to 1) Waveplayer will output more status messages over the UART interface when various events occur, such as events triggered by buttons, .wav file statistics when starting playback, and so on. Unless Waveplayer is intended to be controlled from some external microprocessor and it is desired to reduce the amount of data output this value can safely be left enabled (set to 1).
lowpwr	0	1	1	If disabled (set to 0) Waveplayer will never enter its low-power mode, and instead maintain power to its analog blocks at all times. If enabled (set to 1) the analog sections will be de-powered when playback is inactive. Entering and leaving low-power mode may cause a loud "pop" on the output, see section Usage for more on this.
volume	-100	0	-100	Sets the output playback power, expressed in decibels relative to full power amplitude. IE. a volume of -40 will result in an output volume 40dB below the maximum output volume. The actual output volume from an amplifier or similar will depend on the amplifier's internal gain, the loudspeakers used, and other external parameters.
c_vol_en	0	1	0	Enables (set to 1) or disables (set to 0) external analog volume control as described in section Usage.
shuffle	0	1	0	Enables (set to 1) or disables (set to 0) shuffle playback mode. If enabled the *next* command will randomly select a new track to play, while if disabled a track will be chosen as described in the summary of the *next* command in section Valid commands.
autoplay	0	1	0	If enabled (set to 1) Waveplayer will automatically start playback of the next track when playback of a track finishes. If disables (set to 0) Waveplayer will not start a new track on completion.
btn_pause	-1	11	-1	When triggered is equivalent to calling the *pause* command. A value of -1 signifies that the trigger is disabled. Values of 0 – 5 signify that a falling voltage flank on INP0 – INP5 will trigger the event to occur. Values of 6 – 11 signify that a rising voltage flank on INP0 – INP5 will trigger the event to occur.

PARAMETER	MIN.	MAX.	DEFAULT	DESCRIPTION
btn_resume	-1	11	-1	When triggered is equivalent to calling the *resume* command. See btn_pause parameter for the effect of a given value.
btn_tgl_pause	-1	11	-1	When triggered is equivalent to calling the *playpause* command. See btn_pause parameter for the effect of a given value.
btn_stop	-1	11	-1	When triggered is equivalent to calling the *stop* command. See btn_pause parameter for the effect of a given value.
btn_playnext	-1	11	-1	When triggered is equivalent to calling the *next* command. See btn_pause parameter for the effect of a given value.
btn_01.wav	-1	11	-1	When triggered is equivalent to calling '*play 01.wav*'. See btn_pause parameter for the effect of a given value.
btn_02.wav	-1	11	-1	When triggered is equivalent to calling '*play 02.wav*'. See btn_pause parameter for the effect of a given value.
btn_03.wav	-1	11	-1	When triggered is equivalent to calling '*play 03.wav*'. See btn_pause parameter for the effect of a given value.
btn_04.wav	-1	11	-1	When triggered is equivalent to calling '*play 04.wav*'. See btn_pause parameter for the effect of a given value.
btn_05.wav	-1	11	-1	When triggered is equivalent to calling '*play 05.wav*'. See btn_pause parameter for the effect of a given value.
btn_06.wav	-1	11	-1	When triggered is equivalent to calling '*play 06.wav*'. See btn_pause parameter for the effect of a given value.

Algorithm 1 List all device parameters and configure Waveplayer disable low-power mode.

```
>get
PARAM         |TYPE      |VALUE           |MINIMUM         |MAXIMUM
verbose       |INTEGER   |1          [rw]|0          [r]|1          [r]
lowpwr        |INTEGER   |1          [rw]|0          [r]|1          [r]
volume        |INTEGER   |-100       [rw]|-100       [r]|0          [r]
c_vol_en      |INTEGER   |0          [rw]|0          [r]|1          [r]
shuffle       |INTEGER   |0          [rw]|0          [r]|1          [r]
autoplay      |INTEGER   |1          [rw]|0          [r]|1          [r]
btn_pause     |INTEGER   |-1         [rw]|-1         [r]|11         [r]
btn_resume    |INTEGER   |-1         [rw]|-1         [r]|11         [r]
btn_tgl_pause|INTEGER   |-1         [rw]|-1         [r]|11         [r]
btn_stop      |INTEGER   |-1         [rw]|-1         [r]|11         [r]
btn_playnext  |INTEGER   |-1         [rw]|-1         [r]|11         [r]
btn_01.wav    |INTEGER   |-1         [rw]|-1         [r]|11         [r]
btn_02.wav    |INTEGER   |-1         [rw]|-1         [r]|11         [r]
btn_03.wav    |INTEGER   |-1         [rw]|-1         [r]|11         [r]
btn_04.wav    |INTEGER   |-1         [rw]|-1         [r]|11         [r]
btn_05.wav    |INTEGER   |-1         [rw]|-1         [r]|11         [r]
btn_06.wav    |INTEGER   |-1         [rw]|-1         [r]|11         [r]

>set -p lowpwr 0
PARAM         |TYPE      |VALUE           |MINIMUM         |MAXIMUM
lowpwr        |INTEGER   |0          [rw]|0          [r]|1          [r]
```

Algorithm 2 List all files in the current directory and start playback of '*4.wav*'.

```
>ls
[NAME]                                  [Size (bytes)]
./                                      0
../                                     0
4.wav                                   35996666
Z.wav                                   49836556
a.wav                                   45164208
one.wav                                 41802874
10.wav                                  43468484
>play 4.wav
MSG01: Started playback of file with 2 channel(s), 44100 Hz sample rate, 16 bits
    per sample, and a size of 35996666 bytes.
```

Algorithm 3 Configure Waveplayer to start playback of the next file when a normally open button connected to INP0 is pressed (bringing INP0 to a logic low level) and stop playback when the button is released.

```
>set -p btn_playnext 0
PARAM         |TYPE      |VALUE           |MINIMUM         |MAXIMUM
btn_playnext  |INTEGER   |0          [rw]|-1         [r]|11         [r]

>set -p btn_stop 6
PARAM         |TYPE      |VALUE           |MINIMUM         |MAXIMUM
btn_stop      |INTEGER   |6          [rw]|-1         [r]|11         [r]
```


Sample rate	8kHz, 11.025kHz, 16kHz, 22.1kHz, 32kHz, 44.1kHz, or 48kHz
Bit depth	8-bit or 16-bit
# Channels	1 or 2

Table 4: Supported sample rates, bit depths, and number of channels for WAVE files. All combinations of these parameters are valid and fully functional.

Usage

In order to use Waveplayer, assemble the device and program the microprocessor as per Assembly and connect the device as per Pin Description. As Waveplayer does not draw any significant amount of power there are no significant heatsinking requirements when operating at supply voltages below 20V. At supply voltages above 20V, ensure that the board maintains some power dissipation ability by allowing for some passive airflow.

Next, prepare a memory card with the desired audio track(s). The memory card must be a microSD or microSDHC card, formatted with a FAT12, FAT16, or FAT32 filesystem. The audio files for playback must be formatted as WAVE (uncompressed, LPCM, RIFF audio), with any of the sample rates, bit depths, or number of channels shown in table 4. Typically, the WAVE files should be stored in the root directory (not in a sub-folder) of the memory card. There are a multitude of PC applications that can be used to generate WAVE audio files, one example is Audacity (`http://audacity.sourceforge.net/`). To convert an audio track to a supported format in Audacity, verify that the project rate is set to a valid sample-rate, select the export file option and ensure that WAV (Microsoft) signed 16 bit PCM format is selected.

Depending on the usage case of Waveplayer a configuration file may or may not be needed. If Waveplayer is going to be used in a standalone configuration, where functionality (such as play, pause, next track, and so on) is triggered by external buttons or other sources, see the relevant section below. If however more fine-grained control is needed, such as reading a list of the available files and folders, navigating the filesystem, or triggering playback a specific file (with more than 6 specific files) Waveplayer needs to be externally controlled over its serial interface from a PC or some other device.

Output amplitude

WARNING: Waveplayer can generate a relatively high-amplitude output when approaching maximum volume! Typically this is around 3.9Vpp (compared to 0.7Vpp for most consumer devices). Though damage to the device that Waveplayer drives is unlikely due to the relatively high output impedance (see the section Electrical Characteristics), it is conceivable that high volume levels could cause a large amount of audio distortion in the connected device. If connected to a device that does not support the relatively high output levels generated by Waveplayer the maximum volume level can be reduced by executing the command '*set -max -sudo -p volume -7*', which sets the maximum output amplitude to approximately 0.77Vpp.

Low-power mode

Low-power mode allows for Waveplayer to reduce the power consumption when playback is inactive. If low-power mode is disabled (set to 0) Waveplayer will continuously supply power to its analog sections, both when playback is active as well as when playback is inactive. If idle-state (inactive) power consumption is not an issue this is the recommended setting as there is a significant "pop" on the output when activating the analog subsection if the $\overline{\text{MUTE}}$ line is not used. If low power consumption when playback is inactive is important this parameter may be enabled (set to 1). This will de-power the analog portion of Waveplayer when playback has been inactive for a short period, and re-enable power when playback starts. When disabling/enabling the analog power block there is a significant step in the analog output which will be heard as a loud pop. There is $\overline{\text{MUTE}}$ output that is active low on JP1 that allows for an external device (such as a power amplifier) to disable it's output when Waveplayer has depowered its analog stage. If all audio output is disabled when the $\overline{\text{MUTE}}$ line is held low this can be used to virtually eliminate start/stop "pop" sounds. See section Electrical Characteristics for voltage levels and a diagram of hold times.

To summarize; if idle-state power consumption is important low power mode should be enabled (set to 1) and the $\overline{\text{MUTE}}$ output should be used to reduce the volume of pops when starting/stopping playback. If the $\overline{\text{MUTE}}$ output cannot be used (such as if Waveplayer is connected to an amplifier without a $\overline{\text{MUTE}}$ input) low power mode will generate a relatively high volume pop on start/stop. If this is unacceptable low power mode must be disabled (set to 0). See section Software Interface for information on how to configure low-power mode.

Usage with Cheapamp

When Waveplayer is used with Cheapamp and both are assembled correctly (as described in the section Assembly) some connections are not needed. Power is sourced from the Cheapamp supply (so no connections to V+ IN or GND are needed) and the analog audio output is routed to Cheapamp, so no connections need to be made to OUTR or OUTL. All other connections can be used normally.

Standalone configuration

In this mode all playback control is handled through the general-purpose inputs (input 0 through input 5). These inputs can be controlled by nearly anything — pushbuttons, some other small electronic circuit, relays, a potentiometer, and so on. The following actions are available on either a rising and/or falling signal level on any input;

Pause Pauses the currently playing track (if any) without entering low-power mode. If no track is currently being played there is no effect.

Resume Resumes the currently paused track from the paused location (if any), otherwise no effect.

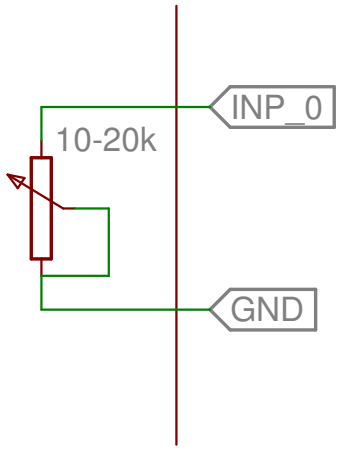


Figure 2: Potentiometer connection for volume control.

Toggle play/pause Toggles the play/pause state (resumes playback if paused, pauses if playing back). If no track is currently being played there is no effect.

Stop Stops playback of the currently playing track (if any) and resets the playback position for the track.

Play next Starts playback of the next track.

Play file *x*.wav Starts playback of a file 01.wav through 06.wav.

Additionally, input 0 can be configured for controlling the output volume by connecting a potentiometer with open-circuit resistance of $10-20\,\mathrm{k}\Omega$ connected as a rheostat between input 0 and ground (see figure 2).

In order to configure Waveplayer, create a plain text file (either Windows or Unix line endings, ISO/IEC 8859-1 / Windows 1252 / ASCII encoding) named "waveplay_config.txt" in the root (base) directory of the memory card. The first line of this file must be exactly "#!waveplay_configfile". All the following lines, with exception of lines that start with the "#" character will be executed by Waveplayer on startup. See the Valid commands section for a list of all valid commands. A sample configuration file is presented below;

```
#!waveplay_configfile
#Turn on verbose mode
set -p verbose 1
#Enable low-power mode. Disable to keep the analog output active at cost
#of power consumption.
set -p lowpwr 1
#Set the default volume. A value of -100 gives minimum volume, a value of
#-7 gives a typical consumer maximum line-level signal, a value of 0 gives
#a signal amplitude of approximately 4Vpp
set -p volume -65
#Enables the external volume control potentiometer. Connect a 10kOhm
#potentiometer from IN0 to ground. Overrides the default volume.
set -p c_vol_en 1
```

```
#Enables random playback order when playing the next audio track.
set -p shuffle 1
#Enables automatically playing the next audio track when the current
#track is finished.
set -p autoplay 1

#All remaining settings allow for binding a button/input to a certain
#event. Setting a parameter to the range of 0 - 5 will cause the associated
#event to occur when INP0 - INP5 goes from a logic high to logic low level.
#A parameter's value in the range 6 - 11 will cause the associated event to
#occur when INP0 - INP5 goes from a logic low to logic high level. The
#following configuration sets up waveplayer for a hardware configuration of
#5 normally open buttons connected to INP1 through INP5, where;
# - Pressing button 1 triggers the next track to be played
# - Pressing button 2 pauses playback, releasing button 2 resumes playback
# - Pressing button 3 stops playback
# - Pressing button 4 triggers playback of the file "01.wav" (if it exists)
# - Pressing button 5 triggers playback of the file "02.wav" (if it exists)
set -p btn_playnext 1
set -p btn_pause 2
set -p btn_resume 8
set -p btn_stop 3
set -p btn_01.wav 4
set -p btn_02.wav 5
#The following functions are not used in this example and are left unused in
#this example btn_tgl_pause, btn_03.wav --- btn_06.wav
#See the documentation for more thorough descriptions of how these event
#bindings work. Note: The last line to parse must have a newline character
#in order to be parsed.
```

External control configuration

In this configuration all commands are externally driven through the UART interface (UART RX and UART TX terminal connections). Waveplayer uses a standard 8N1 (8 data bits, no parity, 1 start bit, 1 stop bit) serial communication scheme with 0V / 3.3V voltage levels at a baud rate of 57600. All commands and parameters listed in the section Software Interface are available for use. Note that software flow control is used by Waveplayer, and XOFF packets are sent whenever the device starts or stops playback of a file[4]. Waveplayer will not accept any commands until start-up is completed. This time varies depending on the structure of the memory card, but can be detected by the presence of a CLI-ready symbol "$>$" being printed.

[4]These packets are required as playback start/stop requires changing the operation frequency, and therefore some UART peripheral parameters.

Bill of materials

Note; A value of 4n7 corresponds to a value of 4.7n, and in the case of a capacitor corresponds to 4.7 nF. The suggested part number is only that — a suggestion — and may be replaced with any other equivalent matching the specifications listed under value, rating, and type.

Component name	Value	Rating	Type	Suggested Part No.
C1,C2,C17,C21,C26	470n	16 V	X7R ceramic, SMD 0805	CC0805KKX7R7BB474
C11, C12	27p	50 V	NP0/G0G ceramic, SMD 0805	C0805C270J5GACTU
C16	1u	50 V	X7R ceramic, SMD 0805	C0805C105K5RACTU
C3,C8,C18	820p	50 V	NP0/G0G ceramic, SMD 0805	MC0805N821J500CT
C4,C5,C9,C14,C15,C19,C20,C22,C23,C24,C25	10u	16 V	X7R ceramic, SMD 0805	MC1206B106K160CT
C6,C7,C10,C13	3n3	50 V	NP0/G0G ceramic, SMD 0805	GRM2165C1H332JA01D
D1	500 mA	40 V	SOD-123 Schottky diode	MBR0540T1G
D2, D3	-	3.3 V	Quad ESD suppression diode	PESD3V3L4UG
F1	200 mA	30 V	1210 PTC fuse, 200mA hold current	MF-USMF020-2
IC1	-	-	Dual RRIO SO-8 audio op-amp	ADA4692-2ARZ
IC2	-	-	XMEGA32E5-AU	ATXMEGA32E5-AU
IC3	5 V	100 mA	LDO regulator in TO-252	LM2931DT-5.0G
IC4	-	-	PCM1780DBQ Audio DAC	PCM1780DBQ
IC5	5 V	1 A	SOT23-6 load switch	TPS22929DDBVT
IC6	3.3 V	100 mA	LDO regulator in SOT-89-8	MCP1700T-3302E/MB
J1	-	-	2-row 6-way 2.54mm pitch header	2213S-06G
JP1	-	-	1-row 13-way 2.54mm pitch header	MC34735
L1,L2,L3,L4,L5	-	100 mA	SMD 0805 ferrite bead, $> 30\Omega$ @ 1MHz	WE-742792097
LED1,LED2	-	10 mA	SMD 1206 LED	KPT-3216SGC
Q1	4.096 MHz	-	Crystal in HC-49S package	9C-4.096MAAJ-T
R1,R3,R10,R12	6k8	1%	1/4W metal-film through hole resistor	MFR3 6K8 FC
R19	1k	1%	1/10W SMD thick-film 0603 resistor	CRCW06031K00FKEA

Component name	Value	Rating	Type	Suggested Part No.
R2,R6,R11,R15	6k8	1%	1/10W SMD thin-film 0603 resistor	RT0603FRE076K8L
R9, R18, R20, R21	6k8	1%	1/10W SMD thick-film 0603 resistor	CRCW06036K80FKEA
R4,R13	2k2	1%	1/4W metal-film through hole resistor	MFR3 2K2 FC
R5,R14	100R	1%	1/4W metal-film through hole resistor	MFR3 100R FC
R7,R16	2k2	1%	1/10W SMD thin-film 0603 resistor	RT0603FRE072K2L
R8,R17	100R	1%	1/10W SMD thin-film 0603 resistor	RT0603FRE07100RL
RN1,RN4,RN8	10k	5%	Quad resistor array in SMD 1206 package	CAY16-103J4LF
RN2,RN3,RN5,RN6,RN7,RN9,RN10	100R	5%	Quad resistor array in SMD 1206 package	CAY16-101J4LF
X1, X3	-	-	5MM pitch terminal block	CTB5202/6
X2	-	-	MOLEX 500901-0801 micro-SD connector	500901-0801

Mechanical Description

Waveplayer consists of a PCB and associated components, with a finished size of 50mm by 50mm, with a build height of 14mm, limited by the terminal blocks X1 and X3. Mounting holes are listed in table 7 following the coordinate system shown in figure 3. PCB manufacturing requirements are shown in table 6 and should be generally achievable at any PCB house.

Table 6: PCB manufacturing requirements.

PARAMETER	REQUIREMENT	UNIT
PCB thickness	Any (nominal 1.6)	mm
PCB layers	2	-
Copper fill thickness/density (tested)	35/1	µm / oz/ft^2
Trace isolation (minimum)	0.2032/8	mm/mil
Trace width (minimum)	0.254/10	mm/mil
Trace to board edge (minimum)	0.25	mm
Drill to board edge (minimum)	0.4532	mm
Drill diameter (minimum)	0.3	mm
Via annular ring (minimum)	0.2032/8	mm/mil

Table 7: Mounting hole locations.

HOLE DIAMETER [mm]	POSITION X [mm]	POSITION Y [mm]
4.1	5	5
4.1	45	5

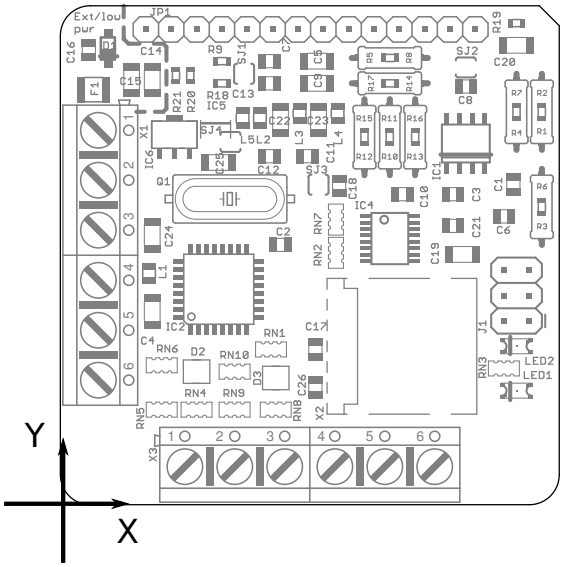


Figure 3: Coordinate system for mounting holes.

Electrical Characteristics

PARAMETER	SYMBOL	TEST CONDITIONS/COMMENTS	MIN	TYP	MAX	UNIT
Input Characteristics						
Typical power input voltage	V_{in}	At input voltages over 20V power dissipation in IC3 may become significant.	6		26	V
Quiescent supply current		Low-power mode enabled		1.7		mA
		Low-power mode disabled		35		mA
		Increase in current per INP0 – INP5 input held <0.1V		0.33		mA
		Increase in current when INP0 held at $V_{CPU}/2$		0.17		mA
Active supply current		Playback active, $R_{load} \geq 10\text{k}\Omega$		80		mA
Input Logic Characteristics						
INP0 – INP5 / UART input impedance		Pullup to V_{CPU}		10		kΩ
INP0 – INP5 / UART logic low voltage threshold					1.02	V
INP0 – INP5 / UART logic high voltage threshold			2.38			V
INP0 – INP5 debounce time		Time INP0 – INP5 must maintain logic level to trigger a detectable logic level change		60		ms
Output Logic Characteristics						
$\overline{\text{MUTE}}$ and $\overline{\text{SHDN}}$ logic high voltage		$I_{source} = 20\mu A$		3.16		V
		$I_{source} = 200\mu A$		1.94		V
$\overline{\text{MUTE}}$ and $\overline{\text{SHDN}}$ logic low voltage		$I_{sink} = 20\mu A$		0.13		V
		$I_{sink} = 200\mu A$		1.36		V
MUTE and SHDN Timing		*Note; see figure 4.*				
$\overline{\text{SHDN}}$ inactive to $\overline{\text{MUTE}}$ inactive	T_{D1}	Only applicable when low-power mode enabled		200		ms
Playback triggered to $\overline{\text{MUTE}}$ inactive	T_{D2}	Only applicable when low-power mode disabled. Note; not shown in figure 4. Similar to T_{D1} except $\overline{\text{SHDN}}$ always kept high.		100		ms

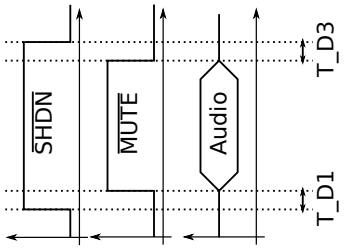


Figure 4: $\overline{\text{MUTE}}$ and $\overline{\text{SHDN}}$ behavior and timing when low-power mode enabled.

PARAMETER	SYMBOL	TEST CONDITIONS/COMMENTS	MIN	TYP	MAX	UNIT
$\overline{\text{MUTE}}$ active to $\overline{\text{SHDN}}$ active	T_{D3}			50		ms
Regulated Voltages						
Digital supply voltage	V_{CPU}	$I_L < 100\,\text{mA}$	3.15	3.3	3.4	V
Analog supply voltage	V_{an}	$I_L < 100\,\text{mA}$	4.70	5	5.25	V
Digital supply to analog supply noise rejection		1 kHz, 10 mV ripple on digital supply	31			dB
		1 MHz, 10 mV ripple on digital supply	44			dB
Output Characteristics						
Output amplitude		Volume parameter set to 0, playing back file with 1kHz sinusoidal 100% amplitude signal		3.9		Vpp
Output impedance				100		Ω
THD plus noise		Volume parameter set to 0, playing back file with 1kHz sinusoidal 100% amplitude signal			0.005	%
Signal-to-noise ratio		Volume parameter set to 0, playing back file with 1kHz sinusoidal 100% amplitude signal	90			dB
Pass-band flatness		$15\,\text{Hz} \leq f \leq 8\,\text{kHz}$			0.1	dB
		$5\,\text{Hz} \leq f \leq 17\,\text{kHz}$			1	dB
Analog Anti-Aliasing Filter Behavior		Anti-aliasing filter is of 2nd order Bessel low-pass type with a cascaded single-pole high-pass (DC-block) filter.				
Anti-Aliasing cutoff frequency		$-3dB$ attenuation		25		kHz
Attenuation at 44kHz				10		dB
Attenuation at 48kHz				12		dB
DC cutoff frequency		$-3dB$ point, $R_{load} > 20k\Omega$		2.5		Hz

Operation

Waveplayer is, at its core, simply a microprocessor with an external SPI memory card for data storage, some I/O for control, and an I2S DAC and analog low-pass filter for analog output. Some notable schematic choices are;

- The analog low-pass filter created by IC1 and associated passive components is dimensioned as a 2'nd order Bessel filter with a cut-off frequency of around 25kHz. A first-order high-pass filter blocks the DC bias and has a cut-off frequency of around 2.5Hz with loads $\geq 20\,\text{k}\Omega$. Another 1st order low-pass filter on the output (R5/R8/C7/R14/R17/C13) clamps any RF interference from propogating from the output to the op-amp, which could cause intermodulation distorsion.
- The DAC is grounded for AC signals to the digital ground through C18, which ensures that the digital input currents can return directly to the microprocessor, rather than through a (potentially) very long loop between AGND and GND. If SJ3 is shorted C18 serves no purpose.
- The analog supply is heavily decoupled and filtered by cascaded ferrite/capacitor filters (L2, L3, L4, L5, C22, C23, C19, C25), whose purpose is to reduce the effects of disturbances on the +5V supply.
- The entire analog supply voltage can be switched on or off at will through the use of IC5. Before switching the analog supply voltage off care is taken to ensure the digital control lines to the DAC are tri-stated. Furthermore, a blanking period between enabling the analog circuitry and driving the output, is used in order to let the analog output stabilize. This parameter (along with others) can be altered at will in the config.h source file.
- All configurable inputs are heavily ESD-protected with a 100-ohm/zener diode T network, which has been shown to be resilient at ESD voltages of up to 15kV as per IEC 61000-4-2, which should serve to be enough for most applications.

The software side of Waveplayer is also relatively straightforward — files are read using Roland Riegel's SD card library (`http://www.roland-riegel.de/sd-reader/`) with very minor modifications. The core clock frequency is switched depending on the sample-rate of the file, and defaults to a low rate (1MHz) when playback is inactive. Note that switching the core clock frequency corrupts any UART communications in progress, which is why software flow control is used. The I2S output is generated by extensive use of the integrated DMA module, which outputs a block of data (the WAV file data) to an UART module. The LRCLK signal is driven by a timer clocked by the UART clock output; this effectively means that the entire I2S bit-stream can be generated completely asynchronously and without any CPU overhead. A single ADC channel is used to read the volume parameter. This channel is only intermittently enabled when playback is inactive in order to reduce power consumption.

For more detail on the software operation, see the source code which is at least marginally commented/documented.

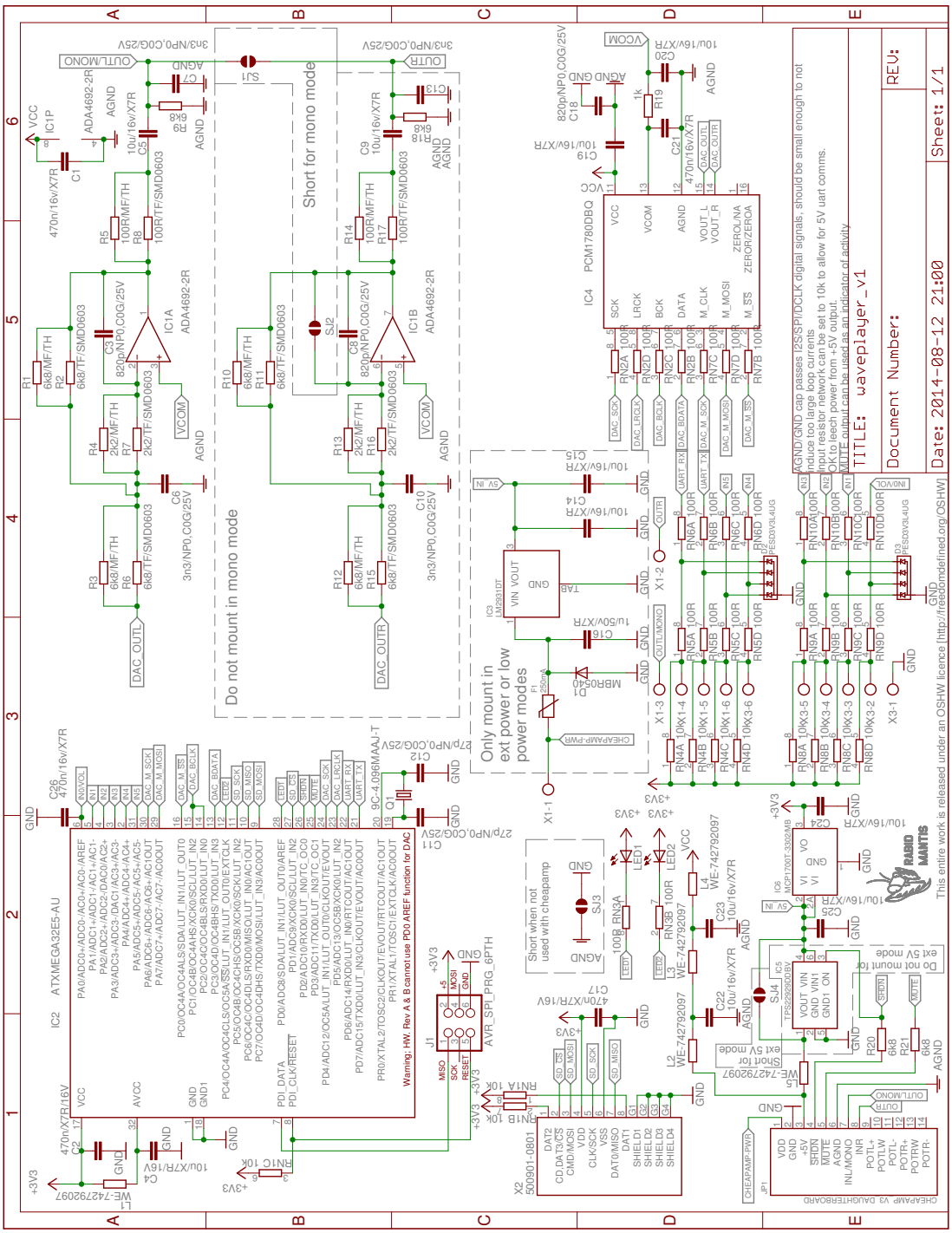
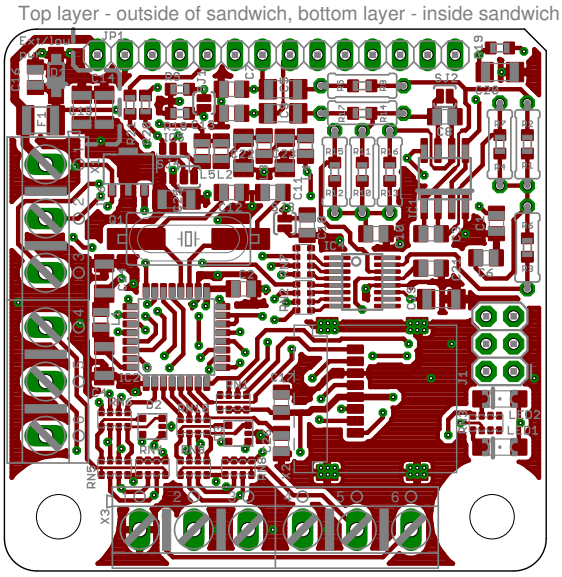
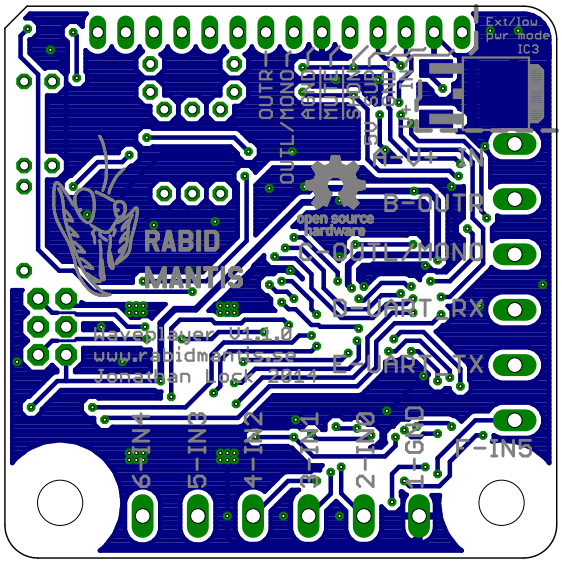


Figure 5: Waveplayer schematic.

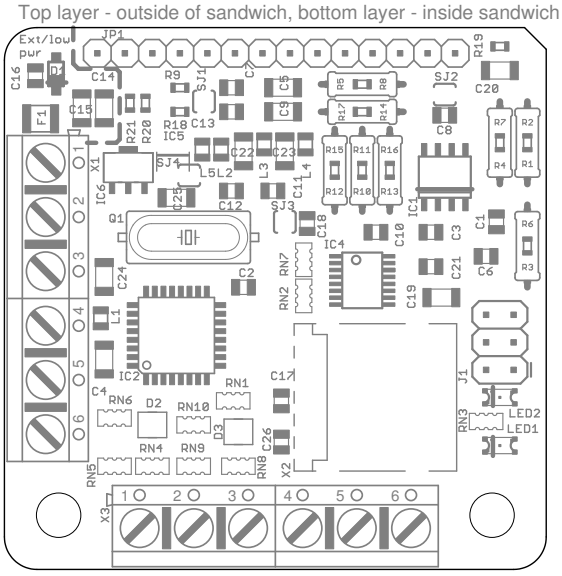


(a) Complete top layer as seen from above. "Sandwich" refers to usage with Cheapamp wherein the entire assembly is referred to as a sandwich, where the inside is the space where Cheapamp is close to Waveplayer, and the outside is the space that is accessible.

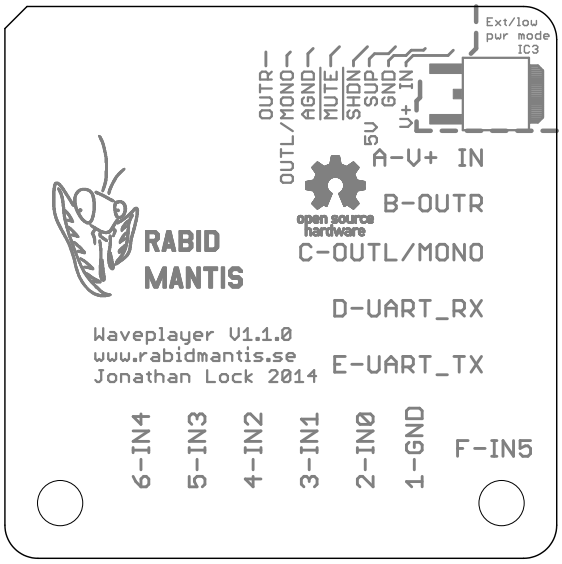


(b) Complete bottom layer as seen from below.

Figure 6: PCB details.



(a) Top layer component outline as seen from above.



(b) Bottom layer component outline as seen from below.

Figure 7: Component placement details.