

---

# Jenkins builds

---

## About Jenkins and the source repository

Jenkins is used as a continual build environment that will automatically build the SoundSystem codebase.

The source repository is available from `svn://gitserver.tribbeck.com:13960/SoundSystem` for read only access. Write access can be granted – please contact Jason Tribbeck for details.

### Code output

Three modules are currently created, each with two variants, and an application which displays information on the installed devices:

#### **SoundSystem**

This is the main module that is the interface for the new Sound System.

#### **DummySoundInterface**

This is a dummy sound interface that simply generates sound data to emulate recorded audio, and requests playback data.

#### **RaspberryPiI2S**

This is an interface for the Raspberry PI to use its I2S interface for both audio capture and playback

#### **DeviceList**

This is a command line application that will output the information on the devices.

### Module variants

Each of the modules are generated with their given name, and also a version with "\_D" appended to it. This version has debug information that is output to !Reporter, which is a logging interface written by Martin Avison.

At the time of writing (early in development), the !Reporter output looks like this:

```

Reporter
20:35:08.97 [Dummy Sound Interface] Allocated 256 byte(s) at 844e53000 (next will be 844e53100)
20:35:08.97 [SoundSystem] Request to register device 'DummyDevice#1' (Dummy audio device #1)
20:35:08.97 [SoundSystem] Using slot 0
20:35:08.97 [SoundSystem] Issuing SoundSystem_ServiceHardwareAdded for 'DummyDevice#1'
20:35:23.57 [Dummy Sound Interface] Freeing dynamic area memory
20:35:23.57 [SoundSystem] Request to deregister device 'DummyDevice#1'
20:35:23.57 [SoundSystem] Deregistering device 'DummyDevice#1'
20:35:23.57 [SoundSystem] Issuing SoundSystem_ServiceHardwareRemoved for 'DummyDevice#1'
20:35:23.57 [Dummy Sound Interface] Dummy sound interface opened
20:35:23.57 [Dummy Sound Interface] Allocated 131072 byte(s) as a dynamic area, located at 844e53000
20:35:23.57 [Dummy Sound Interface] Allocated 256 byte(s) at 844e53000 (next will be 844e53100)
20:35:23.57 [SoundSystem] Request to register device 'DummyDevice#1' (Dummy audio device #1)
20:35:23.57 [SoundSystem] Using slot 0
20:35:23.57 [SoundSystem] Issuing SoundSystem_ServiceHardwareAdded for 'DummyDevice#1'
20:35:47.53 [Dummy Sound Interface] Received service SoundSystem_Initialised 1
20:35:47.53 [SoundSystem] Deregistering device 'DummyDevice#1'
20:35:47.53 [SoundSystem] Issuing SoundSystem_ServiceHardwareRemoved for 'DummyDevice#1'
20:35:47.53 [SoundSystem] Freeing dynamic area memory
20:35:47.53 [SoundSystem] SoundSystem closing down
20:35:47.53 [SoundSystem] SoundSystem opened, allocating 3520 byte(s)
20:35:47.53 [SoundSystem] Allocated 131072 byte(s) as a dynamic area, located at 844e73000
20:35:47.53 [SoundSystem] Allocated 256 byte(s) at 844e73000 (next will be 844e73100)
20:35:47.53 [SoundSystem] Allocated 3520 byte(s) at 844e73100 (next will be 844e73200)
20:35:47.53 [SoundSystem] Issuing SoundSystem_ServiceInitialised call
20:35:47.79 [Dummy Sound Interface] Received service SoundSystem_Initialised 0
20:35:47.79 [SoundSystem] Request to register device 'DummyDevice#1' (Dummy audio device #1)
20:35:47.79 [SoundSystem] Using slot 0
20:35:47.79 [SoundSystem] Issuing SoundSystem_ServiceHardwareAdded for 'DummyDevice#1'

```

This shows the interaction between the various modules during initialisation and finalisation of the modules.

## DeviceList output

An example output from DeviceList is shown below:

### SoundSystem device list

```

-----
Device identifier: DummyDevice#1
Device name: Dummy audio device #1
Max channels: 2
Control code: &20304db8
Control code r12: &20da1934
Device capabilities:
Device supports playback
Device supports recording
Device supports LPCM playback/recording
Device does not support non-LPCM playback/recording
Device supports microphone input
Device supports line input
Device does not support digital input
Device left front mixer not present
Device right front mixer not present
Device centre front mixer not present
Device low frequency enhancement mixer not present
Device left surround mixer not present
Device right surround mixer not present
Device left of centre mixer not present
Device right of centre mixer not present
Device surround mixer not present
Device side left mixer not present
Device side right mixer not present
Device top mixer not present
Device microphone mixer not present
Device line input mixer not present
Device digital input mixer not present
Device formats:
Basic format 0: &00000001 (2-channel 32-bit LPCM) *PLAYBACK* *RECORDING*
@ 48000.00Hz (Buffer = 3840 -> 3840 bytes, default 3840 bytes) *CURRENT*
@ 96000.00Hz (Buffer = 7680 -> 7680 bytes, default 7680 bytes)
@ 192000.00Hz (Buffer = 15360 -> 15360 bytes, default 15360 bytes)

```

DeviceList will show each device's identifier, user-readable device name, maximum number of channels, Control Code pointers and capabilities (using `SoundSystem_EnumerateDevices` and `SoundSystem_GetDeviceInformation`).

It then uses `SoundSystem_GetDeviceFormats` to display a list of all of the sample formats supported, and for each of these, calls `SoundSystem_GetDeviceSampleFormatRates` to find out what sample rates are supported for each sample format. For each of these, it queries the buffer sizes needed for each of the sample rates (default, minimum and maximum).

The current sample format for playback and recording for each device is displayed, along with the current sample rate.

---

# Index

---

## Table of Contents

About Jenkins and the source repository.....	1
Code output.....	1
Module variants.....	1
DeviceList output.....	2