dCS Guide to Computer Audio

Issue 1.10   February 2011

DISCLAIMER:

The information in this manual is supplied in good faith and is intended for general guidance only.

dCS cannot accept responsibility for correcting problems with your computer system / server or for ensuring that all the various combinations of third-party operating systems, streaming software and plug-ins operate correctly with *dCS* products, whether they work with other audio products or not.

Contents

OVERVIEW – THE PRINCIPLES OF COMPUTER AUDIO

Computers have been a part of the audio playback chain since the late 1970s. For years much of the music created around the world has been captured, edited and mastered with a computer. Today a computer in the listening room is no longer restricted to the technophile seeking convenience while sacrificing sonic quality. Macs and PCs are now a viable alternative to high end stand-alone CD transports, CD players and large collections of physical media.

Computer based audio systems are becoming more common and in time may replace instead of supplement the physical media based components in a hifi system. Many audiophiles see this as the end of high quality music reproduction or the beginning of a convoluted fad. Fortunately the opposite is true.

Record labels such as Reference Recordings, HD Tracks, Linn Records, Chesky and 2L have released exact digital copies of their master files in 24/88.2, 24/96, 24/176.4, and 24/196 high resolution formats, meaning that consumers now have the ability to own master quality material and reproduce this high quality music in their homes as never before. Currently, computer based music servers range in functionality, cost and complexity but one of the most important rules for the listener is to never compromise on the resolution of the material they buy. **With a good quality recording, true high end playback is now possible from almost any computer as long as it is configured correctly**.

There is no reason why computer based audio should be more complicated or sound sub optimal compared to a traditional CD transport or CD player. In it's simplest form, a music server consists of a computer, a USB cable, and a DAC.

Anyone capable of playing music through their laptop speakers already has much of the knowledge required to set up and use a music server on a daily basis. This guide can fill gaps in one's knowledge or provide the information for an absolute beginner to become a learned computer audio listener.

The Advantages of Computer Audio

- Convenience – Entire music libraries are now easily accessible
  Many listeners have rediscovered their entire music collections when adding a music server to their systems. People are no longer listening to the last five or ten CDs they've purchased or a limited number of readily accessible discs in the collection. Scanning a wall of CDs is a thing of the past once the discs have been imported into a music server. Finding a specific artist, album, track, or composer is accomplished by a simple search, scroll of the mouse, or tap of the finger.
- High Resolution Support
  The Compact Disc standard developed in the late 1970s and early 1980s is limited to 16-bit / 44.1kS/s resolution. Music servers are capable of a wide range of formats including 16, 24 and 32 bit word lengths and 32, 44.1, 48, 88.2, 96, 176.4, 192 kS/s sampling frequencies.
- Jitter Reduction
  A well implemented asynchronous USB interface can reduce jitter compared to traditional interfaces such as Toslink, S/PDIF and AES/EBU.
- Flexibility
  Computer-based music servers allow tremendous flexibility in the range of supported audio formats, additional data storage, graphical user interface designs, and personal customizations.
- Backup
  A complete music library can be backed up automatically without user intervention. This backup is easily restored in the event of a disaster or simple mishap.
- The Future
  Optical discs as a form of audio distribution are slowly disappearing and music retail stores are closing at an increasing rate. SACD remains a healthy but niche market as it represents a high performance standard. Computer based solutions are clearly the future of audio reproduction and are establishing new references in high end audio quality.

The Disadvantages of Computer Audio

- "PC Phobia"
  Many audiophiles are adverse to computers in the listening room. This PC Phobia stemmed from computers that couldn't live up to audiophile sound quality standards and were difficult to operate. Also, improperly configured computers can damage audio components.
- OS Problems
  The most popular consumer operating systems in the world - Microsoft Windows and Apple OS X - were not created solely to reproduce audio. These operating systems use internal sample rate converters and sound mixers that must be properly configured or avoided to prevent degradation of the playback quality.
- Lack of Technical Knowledge
  The learning curve in computer audio can vary greatly for people due to the huge number of different configuration options avaibale to listenrers.
- Suboptimal Implementations
  There are a plethora of implementations, that can lead one to believe computer audio is incapable of high quality sound. The same method of connectivity (e.g. Firewire or USB) implemented by two different manufacturers can make a night and day difference in sound quality. People often consider this poor sound quality as part of the computer and group all computer based playback into the same category.
- Feature Overload
  The number of options available for computer-based playback are large right now and only increasing. Playback applications, operating systems, and computer hardware all come in many different varieties. All of the options are great for those who understand computer based playback but are overwhelming for newcomers. The key is to find a knowledgeable resource(s) and continue to educate oneself a little bit at a time.
- Other variables (RFI, Noise)
  There are a number of variables affecting sound quality inside a computer. Traditional audio issues like RFI, noise, and power supplies are major concerns with music servers. As time goes on, more of these issues are being dealt with by proper design and new technologies.

Timing and Data

All that is important in digital audio is **timing** and **data** - these two cornerstones of a digital audio system must be considered carefully when building a digital audio system.
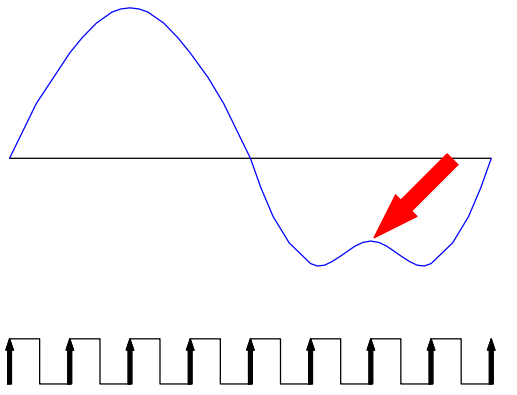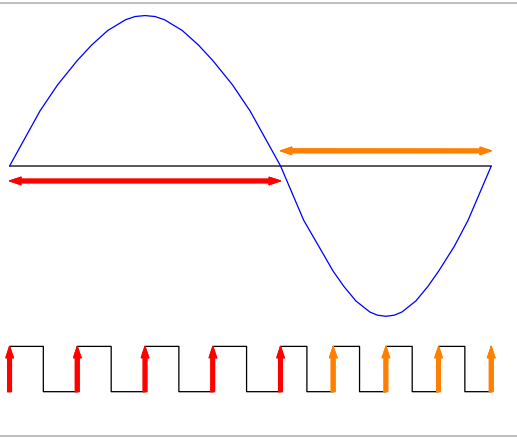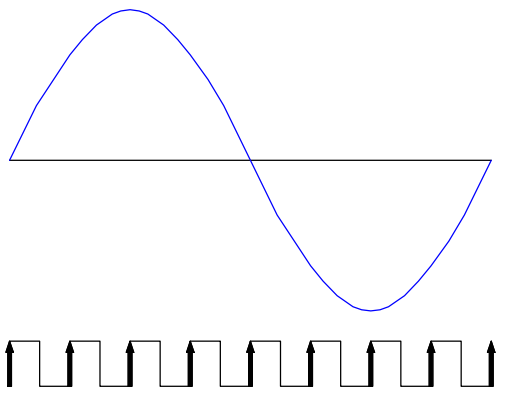
Timing – The stability of the clock in a digital audio system has a major impact on sound quality. This is because jitter is instability or errors in the timebase of a system – usually of very small amounts. This means that if the level of jitter is too large for the system to decode all the bits correctly the system fails because it is working outside its tolerances. Severe jitter usually manifests itself in the form of crackles, pops and is usually of sufficient scale to worry speakers.

The other point at which jitter is important is at the transition between digital and analogue domains – because sampling theory only works properly if the samples are evenly spaced when they re-enter the real world. With a CD (44.1k), the samples are nominally 22.67 microseconds apart in time, and tiny variations in this are the problems that we regard as jitter, causing more subtle effects in playback. For example: Jitter can change a smooth transient into a jagged, uneven one. Also, if there is a pattern to the jitter (for example, power line hum or an audio signal), this can appear as intermodulation distortion in the analogue output.

In a digital system the component most vulnerable to jitter is the DAC. If the DAC generates the master clock for the system, or all of the system components are locked to a very stable, highly accurate master clock, the DAC can largely ignore jitter in the source. In a system that uses a computer or server, clocking is extremely important as the clock generator inside the computer is not designed for music reproduction.

Data is also critical because audio data must move through the components in the playback chain transparently without being altered or modified for any reason other than performance and accurate reproduction.

In digital audio, the waveform is represented as a series of samples (i.e. numbers). Each sample represents a precise amplitude at a precise time. If either are corrupted between the original and the DAC, noise and distortion will result.

	The wrong sample at the right time is the wrong sample. One of the samples is incorrect, resulting in severe distortion. The clock timing is accurate.
	The right sample at the wrong time is the wrong sample. The samples are correct. The clock timing changes part way through the waveform, resulting in severe distortion. (The timing error is exaggerated to make it visible.)
	The right sample at the right time is the right sample. The samples are correct. The clock timing is accurate.

Digital audio equipment should have no trouble keeping the amplitude data correct, provided the computer is properly configured. The timing of the samples is more problematic. The DAC must run at **precisely** the same speed as the source computer, to prevent the data buffer either overflowing or running out of data. This is a much more difficult problem to quantify and solve. Computer based audio via USB interface using proprietary code can improve over older interfaces, such as SPDIF by removing the physical timing link between source component and DAC.

Use of a music server and a correctly configured DAC brings new opportunities for manufacturers to improve upon timing and data flow, by using new interfaces and data movement methods, while consumers have the ability to improve sound quality through the use of different types of computer hardware and software. Well designed and implemented audio components in combination with properly configured computer hardware and software have the ability to sound equal or better than existing playback methods and interfaces.

USB Audio Primer

Universal Serial Bus (USB) is gaining in popularity among audiophiles seeking to connect a music server to their high-end audio system. One reason for this increasing popularity is the ubiquity of the USB interface. USB is available on virtually every computer manufactured in the 21st century. Complexity, confusion, and a unique set of compromises arise when audiophiles involve internal sound cards like the Lynx or RME cards that require installation inside the computer's case.

On the other hand, USB is nearly fool-proof. A USB cable can only connect to a DAC and computer one way and once it is connected, the listener will have sound coming from the computer. Proper configuration will require some fine tuning to get the best sound possible but nonetheless getting sound out of a USB port is quite simple.

Like every other consumer product in audio and elsewhere, not all USB enabled DACs and converters are created equal. By far the most popular USB implementation method uses what's called Adaptive USB mode. The newest USB implementation used by a select few manufacturers is called Asynchronous USB mode. The technical differences between adaptive and asynchronous modes are very large. In addition there are differences between implementations within each USB mode.

Adaptive mode

Adaptive is by far the most common protocol for USB Audio. It works by the DAC adapting it's clock to match the USB frame supplied by the PC.

It is very common because there are many off-the shelf chips that support USB to I^2S or USB to SPDIF, so a manufacturer can easily add a USB input by adding the chip

The drawbacks of using adaptive USB is that the USB frame is generated by the PC, and the jitter and frequency specs are very loose. This means that the output from these devices is heavily influenced by the PC, which is suboptimal in terms of both jitter performance and absolute frequency accuracy. Many implementations are also limited to 16/44.1 and 16/48.

Asynchronous mode

USB is isochronous. This means that the host (Computer) & client (DAC) both know how much bandwidth is available at the outset, so the host can guarantee that bandwidth will be available all the time. USB also natively supports a mechanism by which the rate of data delivery from the host device (PC) can be controlled by the receiver (DAC). This allows the receiving device to feature a high quality clock and/or slave to an external clock source and be immune from the jitter and other inaccuracies of the host PC's clock. Firewire does not provide for this type of feedback, so is less suitable for this type of application. Additionally, USB is more widely available, especially on older PCs, while 1394 is disappearing from new PCs.

Asynchronous USB works by operating a feedback pipe back to the PC. The feedback channel is used to ensure the rate of data to the DAC matches the DAC clock – meaning that the DAC is in charge of the timing of the data, and the PC just has to provide the data at the rate requested by the DAC. The reason asynchronous USB is not used as widely as adaptive is that there are no off-the-shelf solutions – so the manufacturer has to do the design work themselves to implement the USB interface. This obviously is more expensive in design effort but worth the effort in quality and performance terms.

USB 2.0, 3.0 and beyond

USB 3.0 is appearing on new PCs, so the question arises: is USB 2.0 fast enough? In theory, USB 2.0 supports data transfers at up to 480Mb/s while USB 3.0 is specified at 4.8Gb/s. In practice, the speeds achieved by USB are perhaps 2/3 of these theoretical rates, but that is still very fast.

Faster is usually better, but what do we actually need for audiophile-level sound quality? Stereo 24/192 format uses only 9Mb/s, so audio quality is not being limited by even USB 2.0 at present. We can expect evolution towards 32/384 data in the coming years, but even this will not stretch the capacity of USB 2.0. The designers of USB 3.0 have, very sensibly, specified that it includes a secondary USB 2.0 interface, so we can expect USB 2.0 to be popular for some time yet.

THE COMPUTER

The minimum system requirements for computer based audio system are:

1. A Computer (laptop or desktop)
2. Playback software
3. Connectivity between computer and audio component (e.g. USB)
4. Cable (e.g.1 x USB cable)
5. Audio Component with USB input

Comparing these requirements to a traditional high end digital audio system: Requirements 1, and 2 accomplish the job of a CD Transport. Requirement 3 replaces S/PDIF coaxial or Toslink interfaces on the CD transport and DAC while requirement 4 swaps one digital cable for another. The Computer Hardware can be as simple as a small laptop or as grand as a custom built desktop tower-type computer - both laptops and desktops have advantages and disadvantages.

Laptop Computer

Almost any laptop purchased from a major retailer today will work as a music server. To take advantage of the improvements possible with a USB DAC, the laptop must have at least one USB port, preferably more. FireWire ports in addition to USB ports are sometimes preferred as external hard drives should use a FireWire interface when using a USB DAC.

It's also advantageous for a laptop to contain a wired Ethernet port. Wired connections perform much better whether one is accessing music over a Local Area Network (LAN) or remote controlling the laptop as a means to select music for playback.

The minimum amount or memory (RAM) the laptop should contain is two gigabytes (GB). Eight gigabytes is often better as this extra memory allows more flexibility for playback options. Adding more than four gigabytes should not have any negative effects. One reason 2GB can be limiting for a music server is the use of memory playback. Applications such as J River Media Center, Amarra and Pure Music have options to play files back from memory only. These players can load an entire playlist into RAM. This decreases the amount of disk activity and can allow users to close the application's graphical user interface during playback. Laptops with only 2GB of memory can quickly use the entire 2GB once the operating system, playback application, and several high resolution files are loaded into RAM. Once a computer's RAM is full or nearing capacity the computer uses virtual memory. This virtual memory is not memory at all. It's simply a file located on the hard drive containing the data that couldn't fit into the available RAM. Virtual memory is much slower than RAM and increases disk activity greatly. Neither of these deleterious consequences are good for music servers and sound quality.

Many computer operating systems cannot use more than 4GB of memory. All 32-bit versions of Windows XP, Windows Vista, and Windows 7 have a 4GB memory limit. 64-bit versions of Windows Vista and Windows 7 can use between 8GB and 192GB, depending on the version of each operating system, while Mac OS X can use up to 16TB (terabytes) of RAM.

The amount of hard drive space and type of hard drive required vary widely. It's usually a safe bet to purchase the largest internal hard drive one can afford. An uncompressed WAV or AIFF file consumes approximately 10 megabytes (MB) per minute of each track. 1000MB equals 1 gigabyte (GB), and approximately 100 minutes of music consumes 1 GB of hard drive space.

More important than hard drive size is the type of drive. Traditional hard drives use technology based on a 1956 design for IBM accounting computers. These drives all contain spinning platters and a scanning head that reads data as a user accesses that data. A spinning hard drive in a music server must instantly read approximately 40MB of data spread across several platters and stored in a disparate manner on each platter in order to play one track. The main advantage of a spinning hard drive is the large amount of storage available. It is common today for laptops to contain 500GB of hard drive space.

However, over the last couple years Solid State Drives (SSD) have become more affordable and available for use in computers and music servers. SSDs contain no moving parts and access data in an entirely different way with typically lower access times, lower latency, lower power consumption, and completely silent operation. The disadvantage of SSDs is the amount of storage available. 64 to 128GB are common sizes of SSDs available today. Larger SSDs such as 256 GB are currently around 10x the cost of a comparable spinning hard drive while 500GB SSDs are virtually nonexistent for laptops. Like audio components, not all solid state drives are created equal. Two main types of SSD drives are MLC and SLC. The SLC drives are currently preferred for a number of reasons including speed and longevity of the drives. MLC drives continue to increase in performance and longevity and may soon equal that of SLC drives.

For the best of both technologies, choose a solid state internal drive for the operating system and applications, then use an external hard drive or NAS for music storage

Advantages of a Laptop

- Small size.
- More aesthetically acceptable.
- Built-in mouse, monitor, and keyboard for control of all applications.
- Portable.

Disadvantages of a Laptop

- Severely limited hard drive drive space.
- Lower spec for the money compared to a desktop or mini desktop.
- Limited number of USB/Firewire ports.

Desktop Computer

Many of the requirements and recommendations for laptops hold true for desktop music servers. USB port(s), wired Ethernet, RAM, and hard drive space recommendations are generally the same. Desktops do differ in terms of hard drive configuration. Many desktops can house two to four hard drives internally. This allows one to combine several smaller drives at a cheaper price than fewer larger drives. This can be very important if using SSDs as larger drives may not be available. Four 128GB SSDs can be combined in a desktop as one large 512 GB SSD, although this is not recommended. In this configuration, one drive failure destroys all data on all drives.

There are other options to combine hard drives in a desktop using redundancy. RAID is an acronym for Redundant Array of Independent Disks. The popular RAID options RAID 0, RAID 1, and RAID 5 will be briefly addressed later in this guide.

A common desktop hard drive configuration is one hard drive for the operating system and applications and another hard drive that contains the music collection. The drive containing the music can be 2 terabytes (1,000GB or 1,000,000MB = 1TB) and the drive containing the operating system can be much smaller. Operating system drives around 160GB are common. Desktops also allow for the use of an internal audio card sometimes called a digital I/O interface.

Two common audio cards used in high end audio music servers are the Lynx AES16 and RME HDSP AES-32. These audio cards support all sample rates from 44.1kHz to 192kHz. Both the Lynx and RME cards support dual wire AES configurations that are required by some components like *dCS* DACs.

Both cards support use of an external clock such as the *dCS* Puccini U-Clock, but the RME card currently supports clock multiplication (e.g. a base clock rate of 44.1kHz can be used to synchronise 44.1, 88.2 or 176.4kHz files) whereas the Lynx card requires a clock frequency that matches the music file's sample rate.

Advantages of a Desktop

- Highly expandable.
- Large hard drive capacity.
- Ability to integrate internal audio cards.
- Can be customized to operate silent and fanless without moving parts.

Disadvantages of a Desktop

- The size of a desktop computer is often considered unsightly.
- Desktop PCs can be too noisy for a critical listening environment.
- Desktop PC requires mouse, monitor, and keyboard or use of another portable computer (e.g. laptop) for many tasks. This means many moving parts that require power and space. There has been an increasing trend toward using tablet devices and smart phones with VNC applications for some level of remote control.
- More moving parts than a laptop.

Additional Storage Options

The amount of disk space necessary to house a music collection varies widely from person to person. One guarantee is that a person's music collection and disk space needs will continue to grow over the years. A typical album ripped at lossless compression will use approximately 500MB so the easiest way to accommodate a large or growing music collection is through the use of a Network Attached Storage (NAS) device.

In simple terms, a NAS is a hard drive connected to a home computer network or LAN. This network attached storage device is accessible from any music server or computer attached to the network. It allows a single copy of the music collection to be accessed simultaneously by several music servers. The most common NAS devices actually contain two to four hard drives but will appear to the user as a single large hard drive. This type of configuration means one can store a 3TB music collection on a single NAS device. This is not currently possible on a single hard drive.

Another benefit of a NAS device is built-in data redundancy through the use of RAID 1 or RAID 5. Most NAS devices also allow in-place disk upgrades with no loss of data or down time. If a NAS contains four 1TB hard drives and additional space is required, these NAS devices allow the 1TB hard drives to be replaced with 2TB drives (one at a time) very easily without user reconfiguration.

In addition to NAS devices, USB and FireWire external storage units exist that function similar to a NAS. Since these units are directly connected to a music server, only that music server can access the music contained on the hard drives. This type of external locally connected drive can be easier for the user to configure but they are more expensive and frequently don't performance as well as a NAS.

RAID Overview

Redundant Array of Independent Disks is the gold standard used around the world for data protection through redundancy. Using RAID will protect a music collection from loss during certain hardware failures. A major reason why audiophiles should use a form of RAID when storing music is time. The amount of time it takes to rip an entire library of CDs and possibly convert vinyl albums to digital files can be enormous. Once a collection of CDs has been ripped to a music server, it is common to spend several hours cleaning up the information about each CD such as album art, composer, and artist. Going through this whole process more than once is undesirable and may be very expensive if a CD ripping service is used.

Three types of RAID configurations most commonly considered for music storage.

- RAID 0 is **not** recommended for storing a music collection. This form of RAID is frequently used in high performance gaming computers. Using RAID 0 for a music collection will eventually lead to loss of the music collection.
- RAID 1 is referred to as drive mirroring. This form of RAID often uses two identical hard drives with the exact same data on each. When one drive fails no data is lost.
- RAID 5 is commonly used with more than two hard drives. This form of RAID uses an algorithm to spread the data across all the hard drives with parity information. When a single drive fails, no data is lost. Data from the failed drive is available from the remaining drives until the failed drive is replaced.

Backup

Backup of your music collection is essential. Using RAID is NOT a form of backup rather it is redundancy. RAID will not protect files from deletion, corruption, disaster, or user error. For example, using RAID 1 (mirroring), if an album is accidentally deleted from a hard drive, that deletion is instantly mirrored to the other hard drive(s). A copy does not remain anywhere that's readily retrievable. Regular data backup provides a copy of the deleted album one can restore. Ideally backups should be made to an external hard drive or NAS device that can be moved to another building. This can save your music collection in the event of a fire or similar disaster. If an off-site location for the backup is not possible, backing up to an external hard drive that is only connected to the music server during the backup process is recommended. Some external hard drives available today have single button backup utilities that require little if any user interaction once configured.

A popular and very easy backup method for Mac OS X is through the use of Time Machine and a Time Capsule. Time Machine is an application included with Apple's OS X operating system. The application automatically backs-up the data to a device called a Time Capsule. The Apple Time Capsule contains a hard drive and is connected to a home network (LAN). This backup can be done via wired Ethernet or wireless 802.11n. No user intervention is required to backup data several times each day. The Time Machine application can also backup to a locally attached hard drive or a hard drive attached to an Apple Airport Extreme. Restoring data from a Time Machine is easilyd done through a point and click user interface. The maximum capacity of a Time Capsule is 2TB.

Current Windows based backup solutions are either more limited in size or ease of use. The Seagate Replica uses the same concept as the Time Capsule but is limited to 500GB (or 0.5TB). Seagate also offers its FreeAgent line of products that include backup software for Windows. Running backup software in the background while using a music server is not recommended. There are a limited number of one-touch backup solutions for Windows. These external drives offer a single button on the drive housing that launches the backup process. This is a manual operation but may allow the user to run a backup only when needed.

Which Operating System?

One of the most common questions about computer based audio is which operating system to select. There is no "best" operating system for a music server. Great sound quality can be achieved on Windows, OS X, and Linux as long as the proper configuration is in place. Computer audio beginners should select an operating system they are comfortable using. Ideally a music server should not be used for other tasks like browsing the Internet, composing documents, or playing games. Music servers work best without many of the standard operating system features turned on. Anti virus software should never be installed on a music server. These applications are very resource intensive and can interfere with audio playback.

Apple OS X is the easiest operating system to configure for bit-perfect digital audio output. The current version, 10.6, is also named Snow Leopard. Snow Leopard is a 64-bit operating system. iTunes is installed as part of the operating system and will satisfy many users for its easy of use and library management. The number of music playback applications is fairly limited for OS X. The operating system itself is very stable and will usually run without problems when used as a music server. OS X has a built-in remote control feature for total control over the music server from any location on the network (LAN). Apple's Remote iPod Touch application also works well to control music playback through iTunes. In 2009 Sonic Studio released an OSX application called Amarra that works with iTunes to improve performance and enhance its usability when playing music collections with a variety of sample rates. In 2010, Channel D's Pure Music Player was released, this works in a similar way.

Microsoft Windows is by far the most popular operating system in the world. Windows 7 is the current version and has been received very well by audiophiles. Windows 7 is available in 32-bit and 64-bit as well as a number of different versions. Windows 7 Professional and Ultimate have the remote desktop capability built-in whereas the lesser versions require a third party application for this feature. Both the 32 and 64 bit versions of Windows 7 work well as music server operating systems. The only major advantage to the 64-bit version of Windows 7 is the ability to use more than 4GB of RAM, provided the hardware is also able to accept more RAM. Microsoft Windows XP is also a viable music server operating system. XP is even preferred by many audiophiles for its simplicity compared to Windows Vista and Windows 7. All three Windows operating systems are capable of equally great sound quality. Microsoft Windows also has the most music playback application options of any operating system.

Linux in the form of different distributions such as Ubuntu, Red Hat, and Debian is a bit different than the mainstream operating systems like Windows and OS X. In the long run, Linux-based music severs have incredible potential as they can be customized to do nothing more than play audio. However, Linux is extremely difficult for the average user to install and use, so people unfamiliar with Linux who want to use it as the basis for a music server should seek out a commercially available turn-key Linux based music server. Most popular versions of Linux work very well with USB audio components.

Equipment Required	What To Consider
Operating System	Microsoft Windows or Apple Mac OS X
Computer Hardware	Fanless, silent design, adequate USB or FireWire ports, separate drives for operating system /applications and music storage
Sound Card	Not required for USB audio.
Storage (Internal)	Solid State vs. Hard Disk, space requirements
Storage (External)	Direct connection via FireWire or USB vs. NAS
Playback Applications	Windows - iTunes, MediaMonkey, J River Media Center, Foobar2000 Apple Mac - iTunes with or without Amarra or Pure Music Player
Remote Control	Apple iPad, iPod Touch, iPhone, smart phones, tablets running VNC Infrared possible - requires IR receiver
Backup	Regular data backup is essential. Automated vs. Manual procedures.
Audio Equipment	DAC or Upsampler + DAC with a suitable interface, Amplifier, Speakers, Cables

Optimising Performance

- Run the a laptop from the power adapter to ensure battery charge is not limiting performance.
- The energy saving options vary from one machine to another. Generally, a computer used as a sound server should be set for high performance. For example:

Windows Vista laptop – Go to **Control Panel** > **Mobile PC** > **Power Options**. Set the **Plugged In** power plan to **High Performance**, set **CPU Speed** to **Maximum** and set **Put Computer to Sleep** to **Never**.

Windows XP desktop – Go to **Control Panel** > **Power Options Properties** > **Power Schemes**. Set **Turn Off Hard Disks** to **Never**, set **System Hibernates** to **Never**.

Mac Mini – Go to **System Preferences** > **Energy Saver** > **Sleep**. Set **Put the Computer to Sleep when it is Inactive** for to **Never**, un-check **Put the Hard Disk(s) to Sleep when Possible**.

- Disable or un-install any security programs (virus scanners, spyware checkers, etc.), as these will periodically demand a lot of system resource, causing the playback to stutter or even stop.
- In general, computer with more memory will give better sonic results.

MUSIC PLAYBACK APPLICATIONS – APPLE MAC OS X

iTunes

iTunes is one of the easiest playback and music library management applications available. Obtaining bit perfect playback via iTunes on a Macintosh computer is very easy. However, iTunes by itself cannot correctly play successive files with differing sample rates without the application being closed and reopened after an OS X Core Audio configuration change. If this procedure is not followed, iTunes will automatically down or upsample the outgoing audio signal.

Amarra

Amarra is a separate application that accesses the iTunes library. Amarra was developed by Sonic Studio in CA, USA. The application's core comes from soundBlade the music mastering software. When used with iTunes, Amarra processes the audio signal instead of iTunes and OS X Core Audio. This enables iTunes playback of multiple sample rates without the current iTunes sample rate locking limitation. Many audiophiles report sonic benefits to using Amarra or Pure Music Player in addition to the added convenience of automatic sample rate switching.

Pure Music Player

Pure Music Player from Channel D uses the library section of iTunes only and features automatic sample rate switching similar to Amarra. Pure Music supports native FLAC and files sampled at up to 384kS/s. A 30-day free trial version is available for download, after which a license code must be purchased ($129 at the time of writing). Please see www.channld.com/puremusic/ for more information.

Apple Mac OS X Setup

The audio device settings are located in **Applications** >> **Utilities** >> **Audio Midi Setup** (6th icon on right in top row).

This is a list of audio devices available on a Mac Pro computer connected to a *dCS* Debussy DAC via a USB cable and with an internal Lynx AES16e audio card. The Lynx AES16e card is inserted into a PCIe slot on the Mac Pro's motherboard. The first five devices in the list are all built into the Mac Pro computer. From this screen it is possible to select the audio output device, such as **dCS Debussy**, then set the clock source and the sample rate of the Debussy manually. In many cases manually setting these parameters is unnecessary.
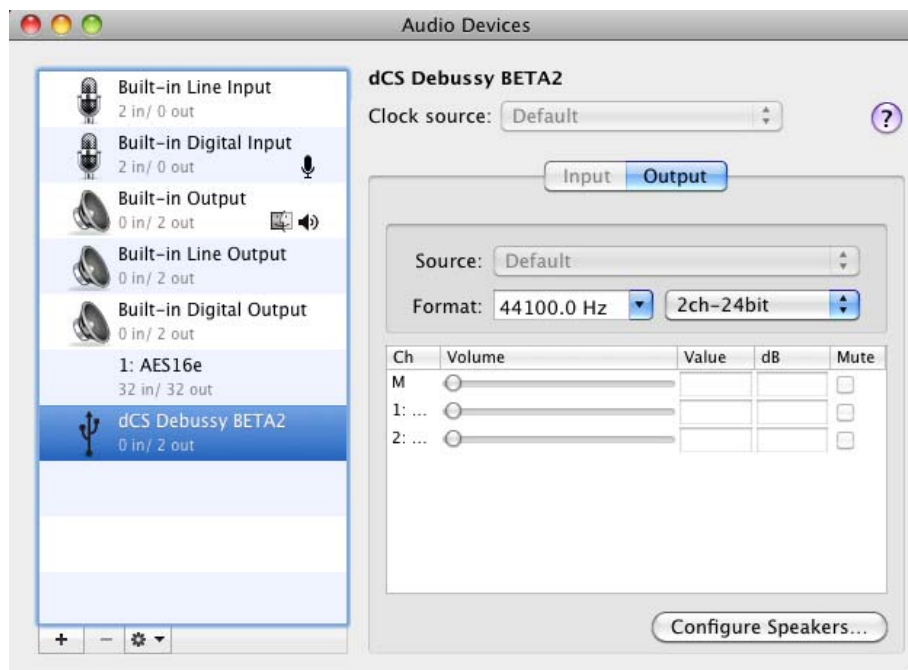
Depending on the system configuration, it may be necessary to specify the correct audio output device. In the picture below, the current audio output device and System sound device is the **Built-in Output**. The Audio device is denoted by the little speaker icon to the right of the device label and the System sound device is denoted by the square Macintosh smiley face. With the System audio device set to **Built-in Output**, all the operating system alerts will only be heard through the computer's built-in speaker. Notice the **Source:** in the picture below is **Internal Speakers**. This setting is recommended as it can be annoying to play the operating system's sounds through a hi-fi system when listening to music.
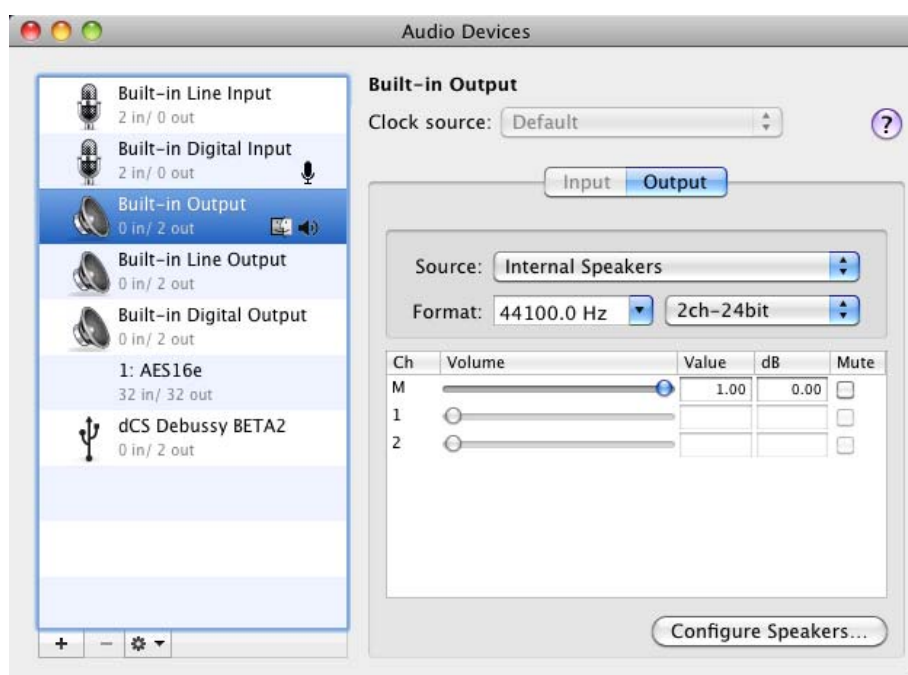
Changing the Audio output device to the Debussy, and the System sounds output device if necessary, is very simple. Select the **dCS Debussy** in the list of devices. The Debussy should then be highlighted blue. After selecting the Debussy, right-Click or hold down the Control key while clicking the **dCS Debussy** again to bring up the context menu displayed below. Select **Use this device for sound output** to set the Debussy as the default audio output device. It is recommended to set the **Format:** to **44100.0 Hz 2ch-24bit** after setting the Debussy as the default audio output device. If the Debussy is set as the System sounds output device, denoted by the square smiley face, the same procedure can be used to change that configuration to the computer speakers. Select **Built-in Output** in the list of devices on the left, then right-Click or hold down the Control key while clicking the **Built-in Output** again to bring up the context menu displayed below. Select **Play alerts and sound effects through this device** to set the computer speakers as the default System sounds output device.
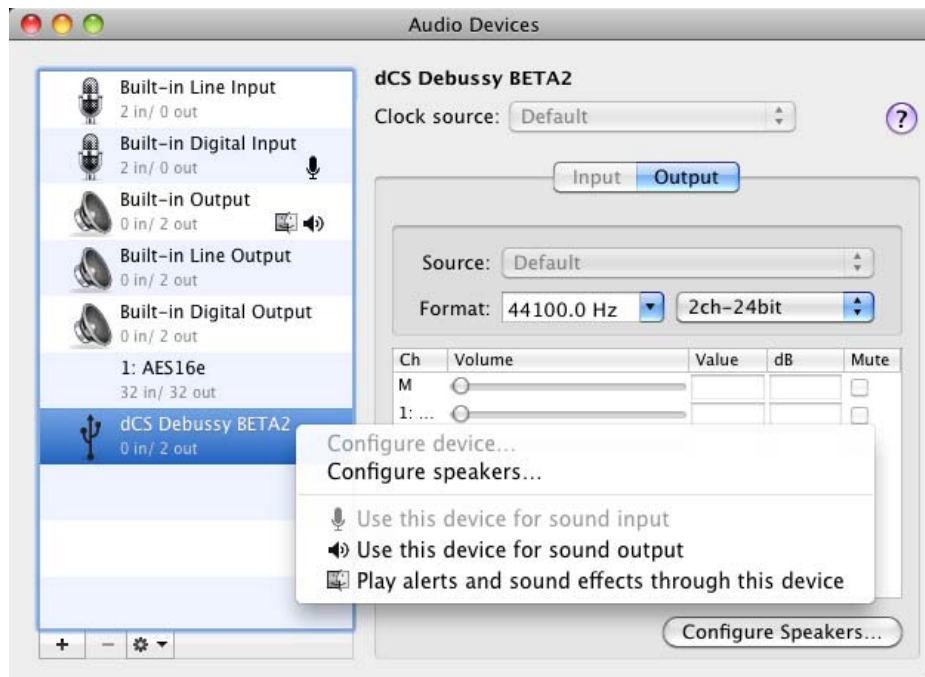
The **dCS Debussy** now has the little speaker icon to the right of the device label denoting it is the default audio output device. The **Built-in Output** has the square smiley face denoting it's the default System sounds output device.

Using iTunes by itself (without Amarra) requires manual sample rate changes. This is the area where such sample rate changes are made. Once the appropriate output device is selected, the **Format** options for that device become available. Since most music available at the time of this writing is 44.1kHz (Redbook CD), it may be best to set the **Format** in this section to **44100.0 Hz 2ch-24bit**. For best results, change the **Format** setting to match the file's sample rate when you play a file sampled at a rate other than 44100.
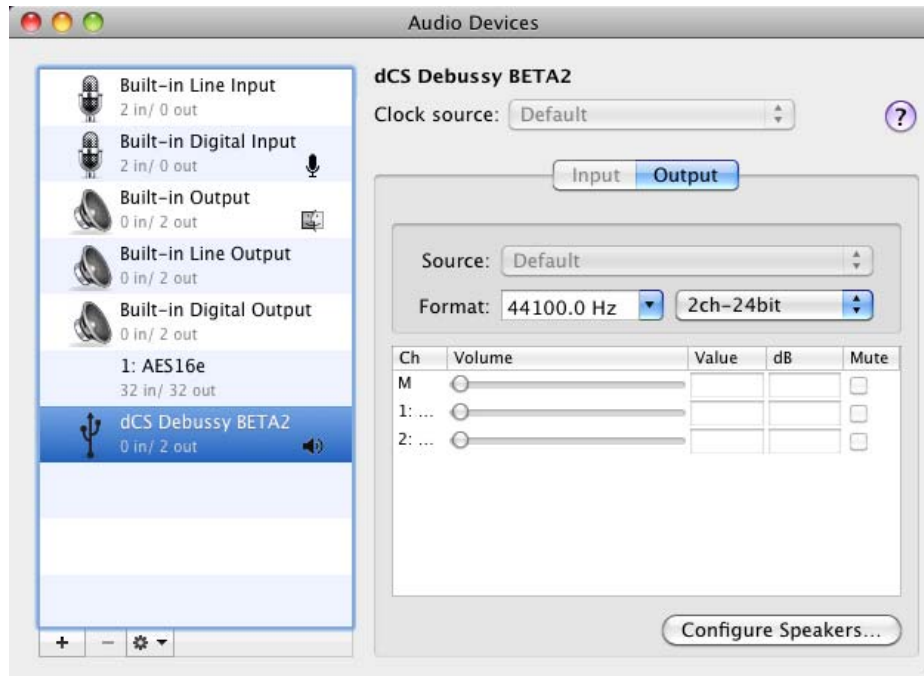
iTunes Configuration

Apple computers are generally supplied with iTunes loaded - please make sure your software is up to date or re-install from www.apple.com/itunes . At present, the iTunes application does not require any changes from the default configuration to enable bit perfect output. Below are screenshots of pertinent iTunes settings for ripping Compact Discs and areas to double check for bit perfect output.

iTunes settings are located beneath the iTunes menu item on the **Preferences** tab.

The **General** tab provides access to iTunes settings for ripping / importing Compact Discs. Select the **Import Settings...** button.

Import Settings is where the file format of ripped CDs is selected.

AIFF & WAV are lossless uncompressed file formats. AIFF supports embedded album art and meta data. WAV does not.

Apple Lossless is a form of lossless compression that supports embedded album art and meta data.

AAC & MP3 are lossy formats that remove data from each track to reduce the file size.

Automatic is the recommended **Setting** for AIFF, WAV, and Apple Lossless files. AAC and MP3 can be customized per user requirements, usually for portable devices.

Two items to check on the **Playback** tab are **Sound Enhancer** and **Sound Check**. To allow bit perfect output, make sure both are disabled by un-checking the check boxes.
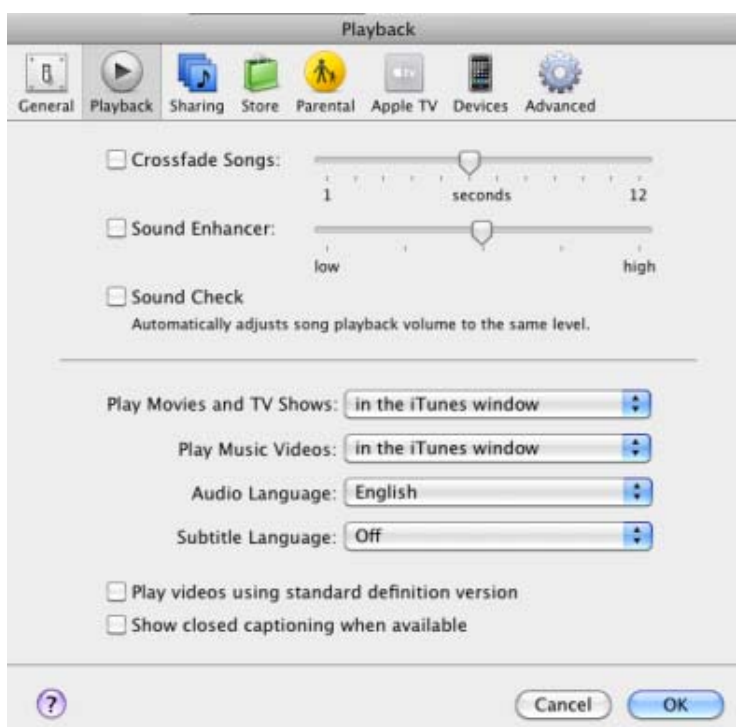
The iTunes **Equalizer** settings can be accessed from the **Window** menu. Make sure the **Equalizer** is either set to **Flat** or the **On** checkbox is unchecked.

The iTunes volume control should be set to the maximum level. This ensures iTunes will not change the audio stream before it is output to the DAC.

Limitations of iTunes with Apple OS X

iTunes with OS X currently, and for the foreseeable future, does not adapt to tracks with differing sample rates unless it is closed and reopened. (The Amarra application solves this problem by processing all the audio through its own software). When iTunes is initially opened, it reads the sample rate set in **Audio Midi**. iTunes then locks on to that sample rate, processing all audio at that initial sample rate regardless of the track's actual sample rate. This can lead to severe sonic degradation. For example if iTunes is opened when Audio Midi is set to 16/44.1 all 16/44.1 tracks will play bit perfect. When a 24/96 track is played iTunes downsamples the track to 16/44.1. The 16/44.1 track is then upsampled by OS X Core Audio to the correct 24/96 sample rate before being sent to the DAC. This conversion is not bit perfect and changes the original data.

The proper way to use iTunes without Amarra:

- Set **Audio Midi** to the exact sample rate of the tracks to be played.
- Open iTunes and play music with that exact sample rate.

To play tracks with a different sample rate:

- Close iTunes
- Set **Audio Midi** to the new sample rate
- Open iTunes and play racks with the new sample rate

We suggest keeping separate playlists for files recorded at different sample rates.

iTunes Remote

The iTunes Remote v2.0 app allows the iTunes library running on an Apple Mac computer to be remote controlled from an iPad, iPhone or iPod Touch through a WiFi network.

- Download the **iTunes Remote** app from www.apple.com/itunes/remote/ into your mobile device.
- Open **iTunes** on the computer.
- Run the iTunes remote app.
- Sync the mobile device and the computer .
- On the remote device, select your music library.

The setup is complete.

Amarra Configuration

The Amarra application, just like iTunes, requires no initial configuration after installation. To make sure Amarra is playing correctly the following items should be checked.

- iTunes volume should be set to the far left level slider location. This disables iTunes volume output completely.
- Amarra volume should be set to the top slider position. This position is designated as 0 because there is no level attenuation (unity gain).
- The correct channel, file format, sample rate, and word length information should be visible in the top window.
- The label of the audio output device, the clock source, and the sample rate should be indicated in the second window.
- The third window simple indicates the HH:MM:SS of the currently playing track.
- The **Power On** button should be illuminated.

NOTE: If Amarra is on (the **Power On** button is blue) it is still possible for the iTunes volume to be adjusted. On rare occasions the iTunes volume slider will get set to full volume by the Amarra application and not returned to volume disabled when playback commences. This leads to both iTunes and Amarra outputting sound for the currently playing track. It is very noticeable as an echo when listening to the track. To resolve this issue slide the iTunes volume back to the far left position to disable the iTunes output.
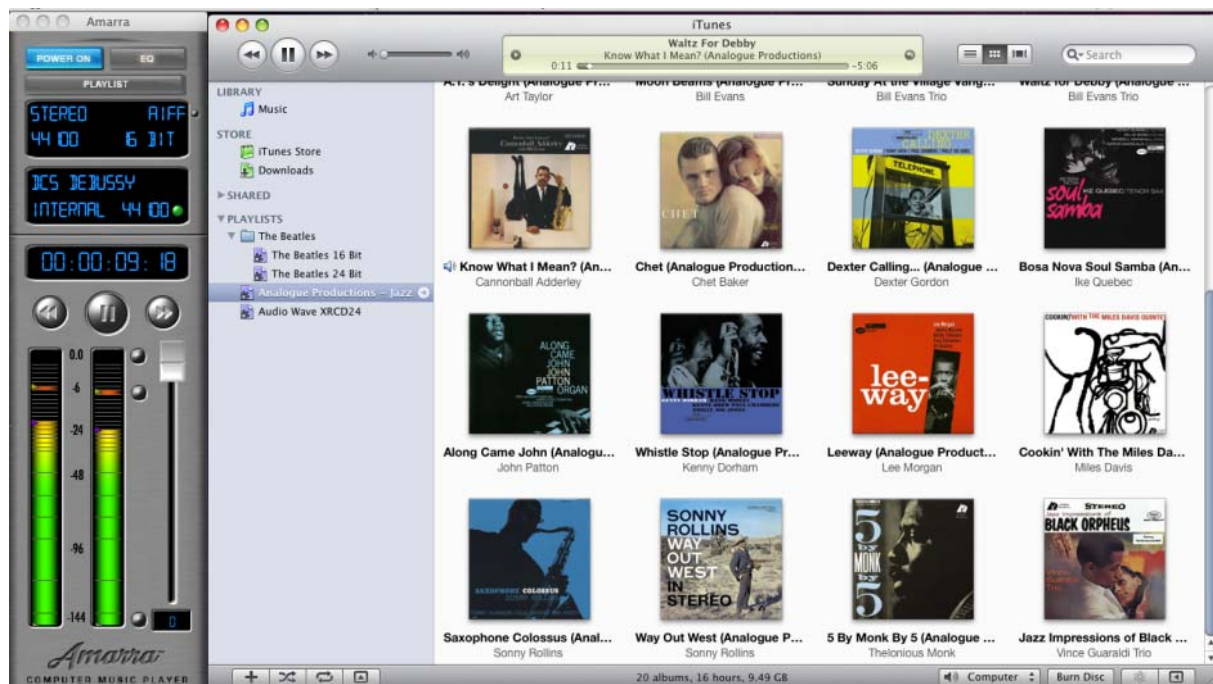
Amarra should automatically set the volume to the appropriate level for bit perfect playback. Check to make certain the Amarra volume level is set to the maximum level and the iTunes volume control is set to the minimum level. The Amarra **Power On** button should also turn blue automatically.

Pure Music Configuration

Download Pure Music from www.channld.com/puremusic/ and install.

For this application, return to the **Apple MAC OS X Setup** section (see page **14**) but do not select the USB DAC in the **Audio MIDI Setup** window – instead select **Built-in Input** and **Built-in Output**.
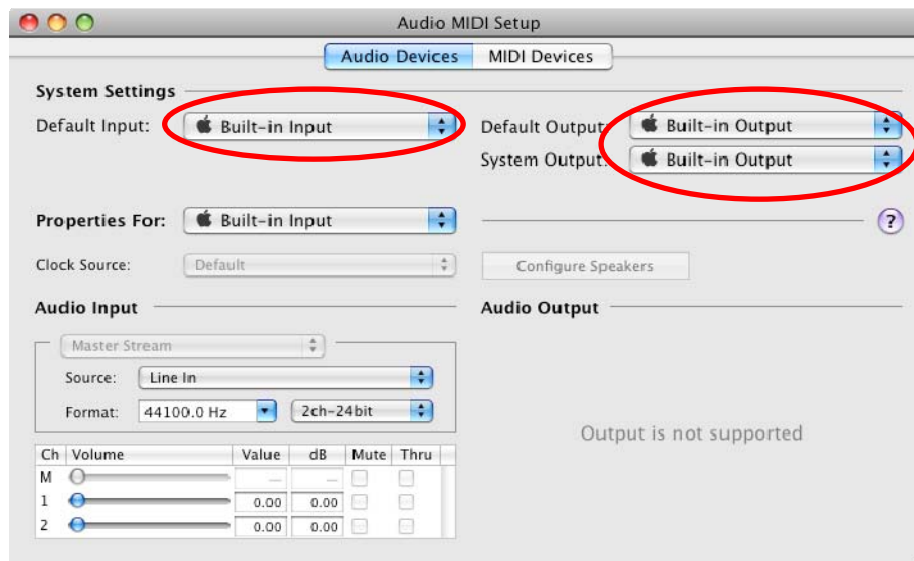
Open Pure Music, click on **Prefs > Advanced Audio Setup**.
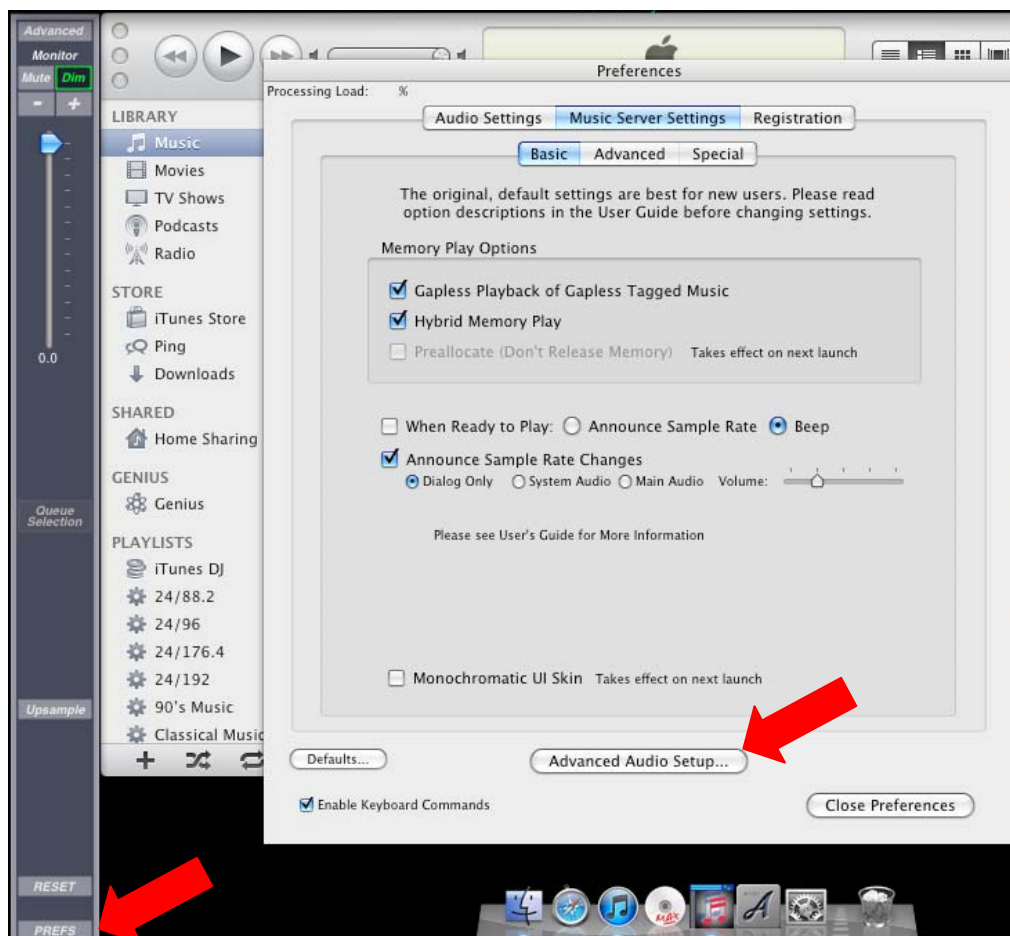
In the **Device Selection** panel, select the USB DAC from the **Output** list. Set the **Sample Rate** to the highest available value (probably **96kHz** or **192kHz**). The **Limit Rate** setting should remain at the default setting of **384kHz**.
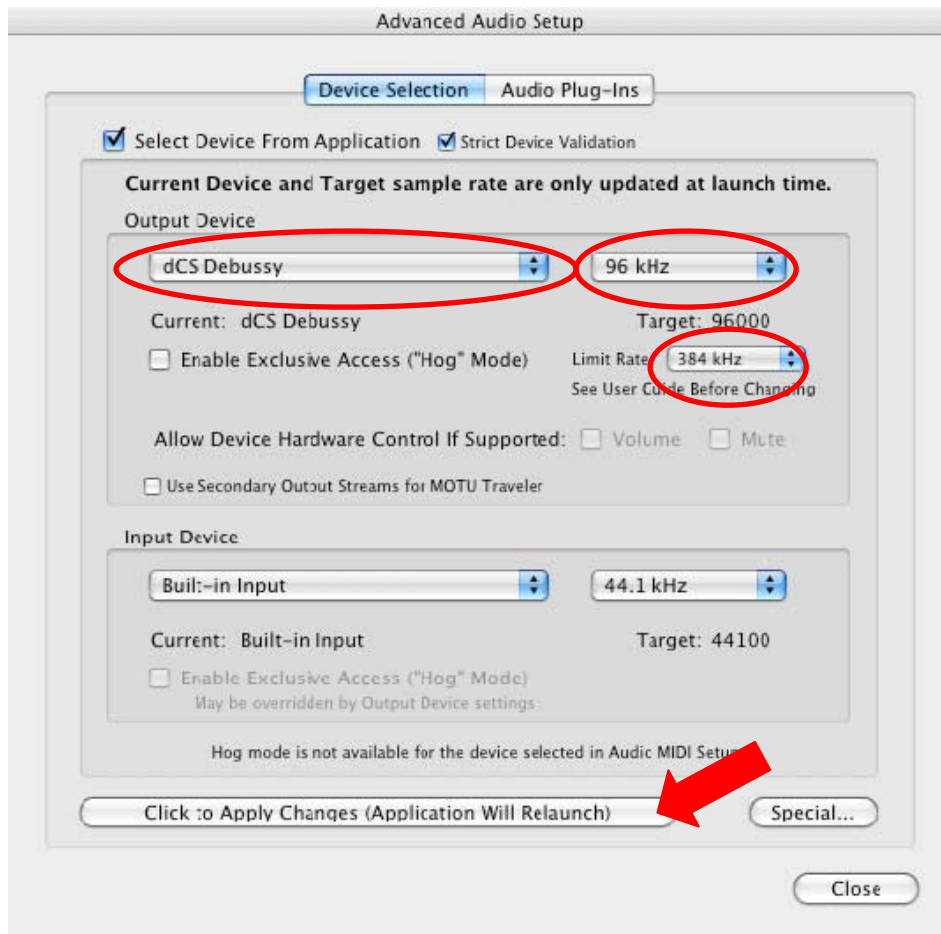
Click on the **Click to Apply Changes** button to accept the changes. Pure Music will re-launch.

Double click a track in the iTunes window to play it.

Suggestions for further improvement:

- If the volume slider is set to **0.0** dB and no DSP options are engaged (plug-ins or upsampling), then the output is bit-perfect (Dithering is automatically bypassed). Plug-ins and upsampling are disabled by default. Set the slider to maximum, then engage the **Dim** function to fix the maximum volume level at 0 dB. Note that you will need to control the volume with the USB DAC or a preamplifier.
- Try using **Memory Play** (plays the track from memory rather than from the hard disk) – click the **Advanced** button in the top left corner, then click the **Memory Play** button.

If you are using the **Memory Play** feature, Pure Music benefits by starting with a "clean slate." If you've been working with other applications on the computer (in particular, any other audio applications / players other than iTunes), a fresh computer restart can be beneficial. Wait a minute or so for the computer to get into a quiet state (total CPU activity below a few percent), as indicated with Apple's **Activity Monitor** application, set to monitor **All Processes**.

- In **Advanced Audio Setup**, try checking **Enable Exclusive Access ("Hog Mode")** - (this allocates maximum system resources to the player.
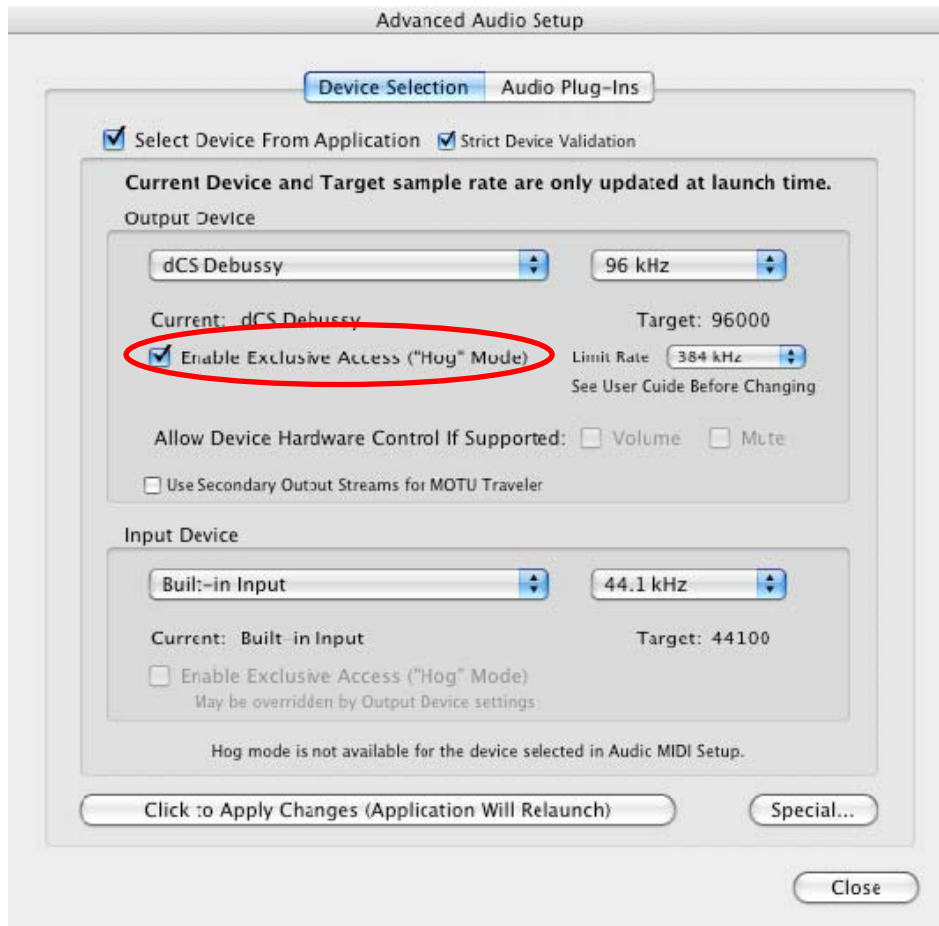
- If you do not need the **Info** and **Metering** displays, disabling them by clicking the buttons in the control bar will release some system resources.

Audio Card Configuration

Using an audio card such as the Lynx AES16 or RME HDSP AES-32 may be preferred by some audiophiles. OSX configuration is described here, but both cards may be fitted to Windows PCs also.

Lynx Studio AES16e

The following instructions detail how to configure a Lynx AES16 in the Mac OS X environment. The Lynx web-site is www.lynxstudio.com. Fit the card inside the computer and install the Lynx software provided.

Double click the **Lynx Mixer** icon within the OS X **Applications** folder.

To enable 24/88.2 playback via AES select **88200** next to **Rate Select** and make sure **SynchroLock** is not checked.
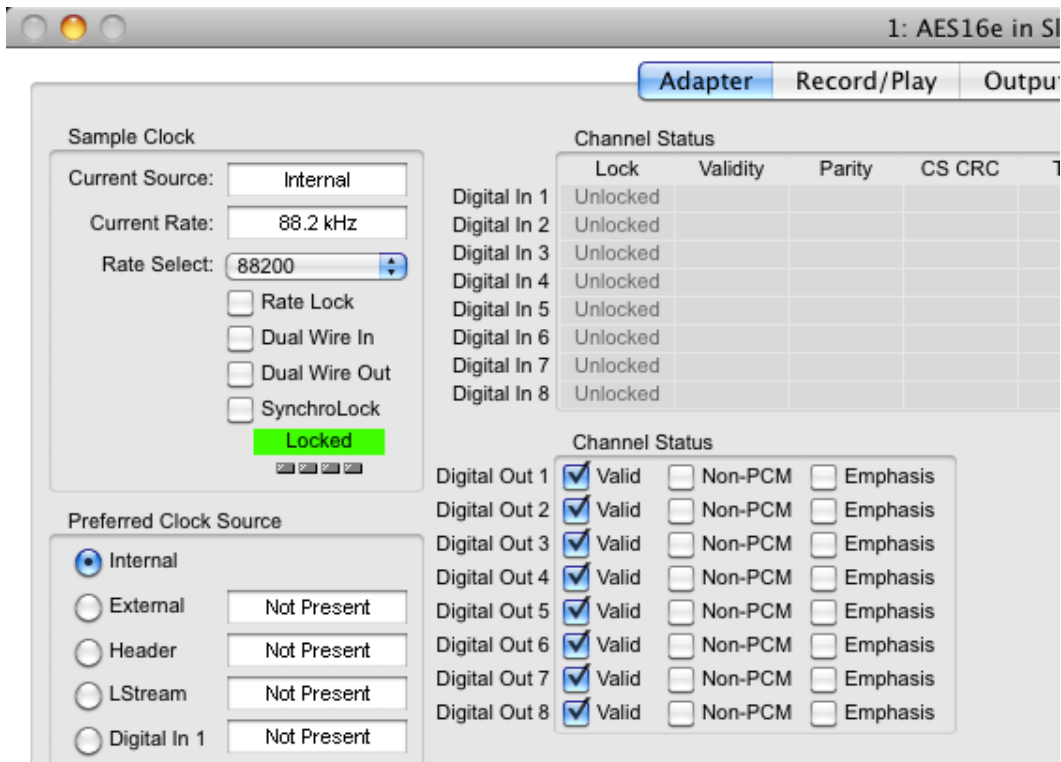
To enable 24/176.4 playback with a DAC that requires dual wire AES for 4x sample rates, select **176400** next to **Rate Select**, make sure **SynchroLock** is not checked, and check **Dual Wire Out**.
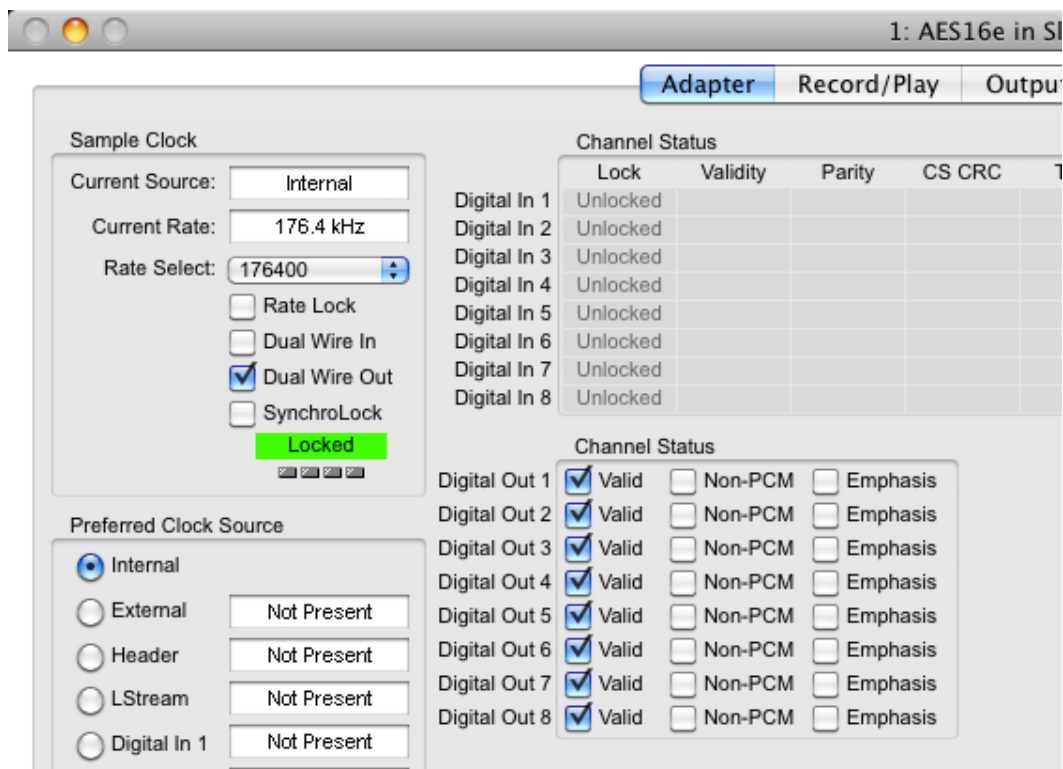
Note that the card can be locked to work clock, but the clock frequency must be the same as the sample rate (e.g. 192kHz clock for 192kS/s data) while the DAC may require a clock at an exact fraction of this (e.g. 96kHz). If you want to lock to word clock, connect suitable word clocks to the card and DAC then check the **SynchroLock** box.

RME HDSP AES-32

The following instructions detail how to configure an RME HDSP AES-32 in the Mac OS X environment. The RME web-site is www.rme-audio.com. Fit the main card (the part with WCL IN & WCL OUT connectors) inside the computer and install the RME software provided.

Probably, you will need to purchase the AES break-out cable to connect to your DAC. The cable available from the RME web-site (part No. "BO25MXLR4M4F1PRO D-Sub Tascam pinout" + length) uses GREEN for Output 1-2 (connect to the DAC's AES1 input) and BLUE for Output 3-4 (connect to the DAC's AES2 input).

Open **Audio MIDI Setup**, set the **Default Output** to **Hammerfall DSP AES-32**, select **Properties for: Hammerfall DSP AES-32**, set the **Audio Output** / **Format** to **192000.0Hz** and **16ch-32bit**. Close the panel.
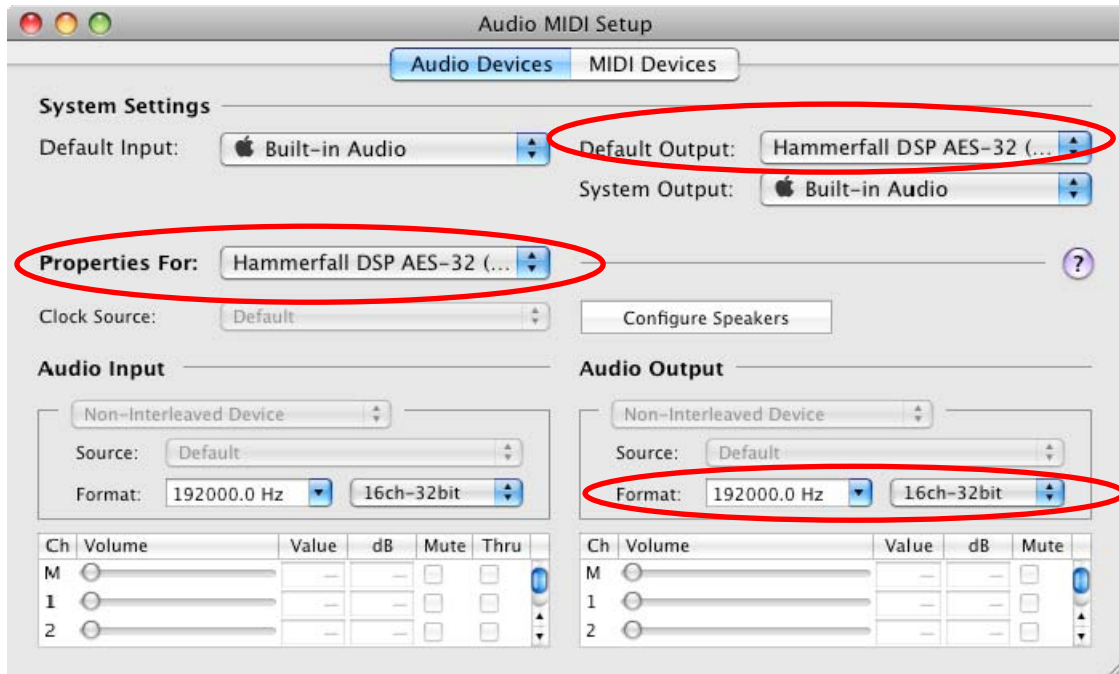
Click on the hammer icon in the dock to open the HDSP Settings panel.

Select the **DDS** tab and ensure the **Active** box is NOT checked.
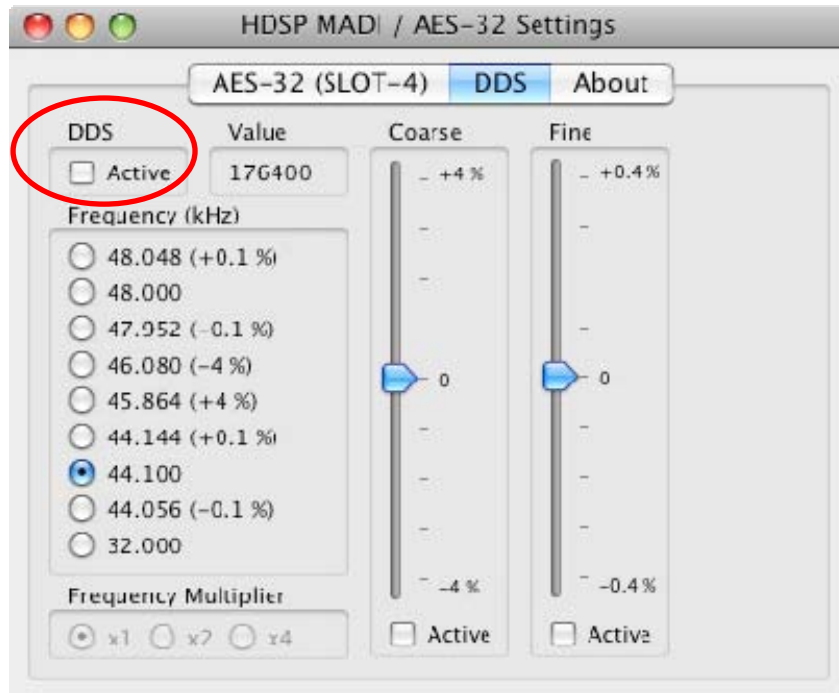
Select the **AES-32** tab and set the **Clock Mode** to **Master**. In the **Output Format** section, set **Double Speed** to **Double Wire**, set **Quad Speed** to **Double Wire** and check the **Professional** box.
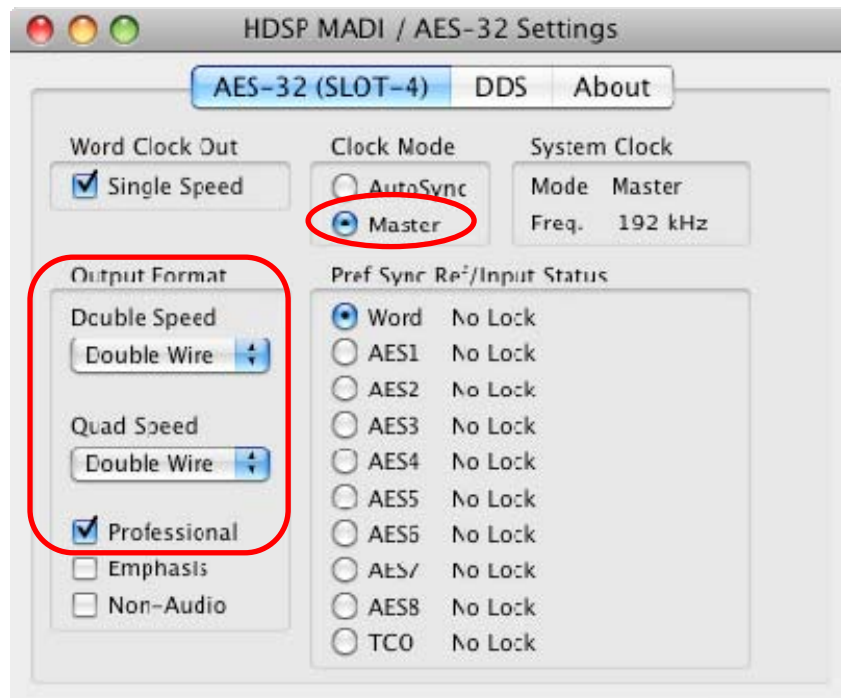
This set-up is easy to use, but it is optimised for files sampled at 192kS/s. Playing files with different sample rates will result in the data being upsampled to 192kS/s, so the data at the output is not bit-perfect.

For bit-perfect results, keep **Audio MIDI Setup** open in the background, each time you play a file with a different sample rate than the previous file, change the **Audio Output** / **Format** setting to match the new file.

Make sure the DAC is set to Dual AES On. Select the Dual AES input for 88.2, 96, 176.4 or 192kS/s data. Select the AES1 input for 32, 44.1 or 48kS/s data.

Adding a Master Clock

If you have a Master Clock in your system, connect one of its word clock outputs to the sound card's WCL IN socket, connect a second word clock output to the DAC's word clock input. Set the Master Clock's frequency to match the file sample rate.

- For data at 44.1, 88.2 or 176.4kS/s, the clock frequency can be 44.1kHz.
- For data at 32, 48, 96 or 192kS/s the clock frequency can be 48kHz.

When you change to a different sample rate, change the Clock is set to a compatible frequency if necessary.

In the HDSP settings panel, set **Clock Mode** to **AutoSync** and select **Word** in the **Pref Sync Ref** section. In the **Output Format** section, set **Double Speed** to **Double Wire**, set **Quad Speed** to **Double Wire** and check the **Professional** box.
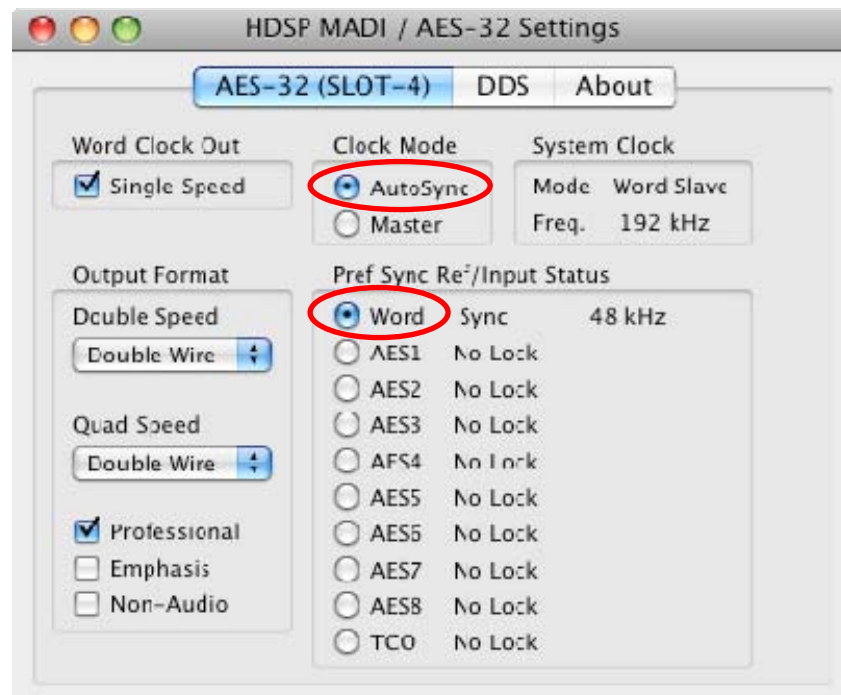
In this example, the file's sample rate is 192kS/s and the Clock frequency is 48kHz.

Close the settings panel.

Useful Links

- **Songbird** - An open source media playback application for Apple Mac, Windows, and Linux. Songbird focusses on open standards and allows the open source community to develop add-ons to the program. The user interface is highly customizable, allowing users to make Songbird look like iTunes or other playback applications. A benefit of Songbird is its ability to natively play FLAC files. iTunes currently does not play FLAC files natively. **http://www.getsongbird.com**

- **Play & Cog** - These two independent applications are very basic and have a minimalist design.
  Play - **http://sbooth.org/Play/**
  Cog - **http://cogx.org/**

- **Max** - The Max application is very good for converting file formats without losing data in the process. Max is frequently used to convert FLAC downloads to a format iTunes can play such as AIFF. **http://sbooth.org/Max/**

MUSIC PLAYBACK APPLICATIONS - MICROSOFT WINDOWS

J River

J River Media Center v15 is one of the most versatile playback applications available for Windows. When configured properly, J River Media Center 15 provides bit perfect digital output. The application can play audio via a number of different Windows plug-ins such as WASAPI, ASIO, DirectSound and Wave_out. A more detailed description of these plug-ins and their function can be found on page 38. J River MC15 also supports multi-room playback in several zones and offers an easy-to-use graphical user interface called Theater View. It is possible to control this application via an IR remote, or iPhone/iPad applications. JRiver MC15 also supports the UPnP/DLNA standards.

Foobar2000

Foobar2000 can be somewhat complicated to use for the beginner. The graphical user interface is very simple by default but can be customized as needed. Foobar has a cult-type following among audiophiles and techies alike. This application is not recommended for beginners.

MediaMonkey

MediaMonkey v3 is a fairly simple playback application that supports a wide range of hardware. It is not as flexible as J River MC14, but is capable of bit perfect output when configured correctly. ASIO is required for bit perfect playback, although other playback methods are available.

iTunes

iTunes v10 is easy to use, it uses the QuickTime player program to interface with Windows. The latest version of QuickTime supports WASAPI, but this does not bypass QuickTime's limiter. iTunes converts audio files to its own internal format.

Supported formats

All 3 applications support a wide variety of music file formats, the most popular formats are listed in the table below.

	J River Media Center v15	Foobar2000 v1	MediaMonkey v3	iTunes v10
FLAC (flac)	✔	✔	✔	Requires extra components
AIF Audio (aif, aifc, aiff)	✔	✔	Requires extra components	✔
CD Audio (cda)	✔	✔	✔	✔
Windows Media Audio (wma)	✔	✔	✔	Requires extra components
Wav Audio (wav)	✔	✔	✔	✔
AAC Audio (m4a, m4p, m4b, aac)	✔	✔	Requires extra components	✔
Monkey's Audio (ape, apl)	✔	Requires extra components	✔	Requires extra components

⚠ This table is based on web sources, we have not checked all the combinations listed.

Windows XP Setup

Windows XP doesn't require any specific settings as an operating system, but there is one setting that is recommended. There is some debate about what exactly happens when an audio device is unmapped in Windows XP. Some explanations concluded an unmapped audio device is the equivalent of using an ASIO driver or bypassing the dreaded Windows XP KMixer. Unmapping an audio device is very simple and can be accomplished following these instructions.

Note: Unmapping a device is only done in Windows XP.

Enter the Windows XP **Control Panel** and open **Sounds and Audio Devices**.
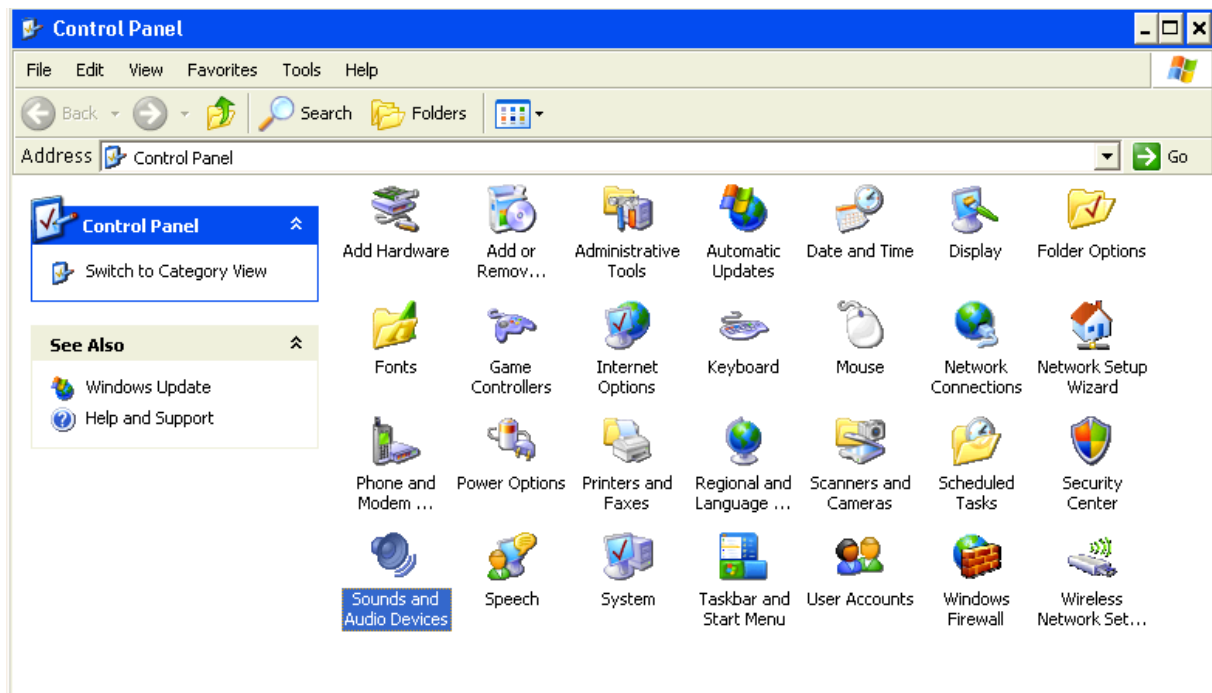
The first tab in the **Sounds and Audio Devices Properties** section will likely display a USB DAC or audio card.
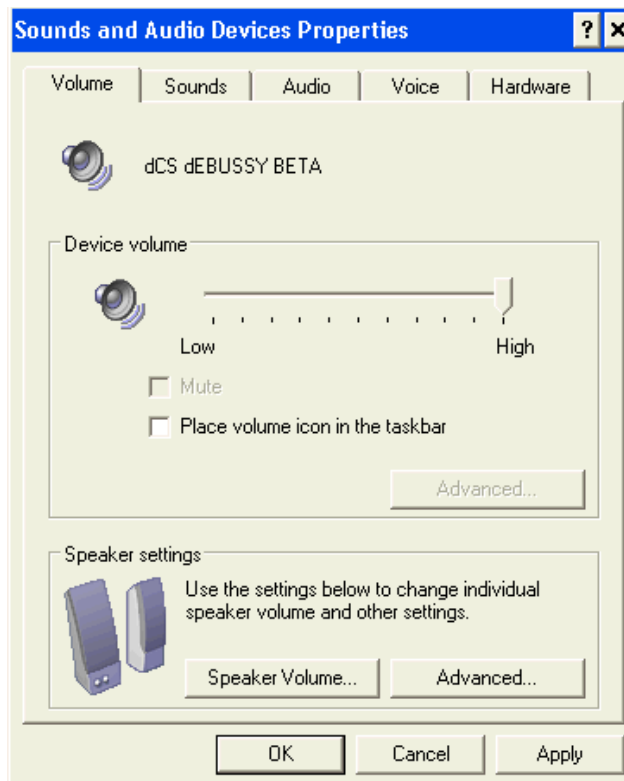
Select the **Hardware** tab, select **USB Audio Device** then click the **Properties** button.
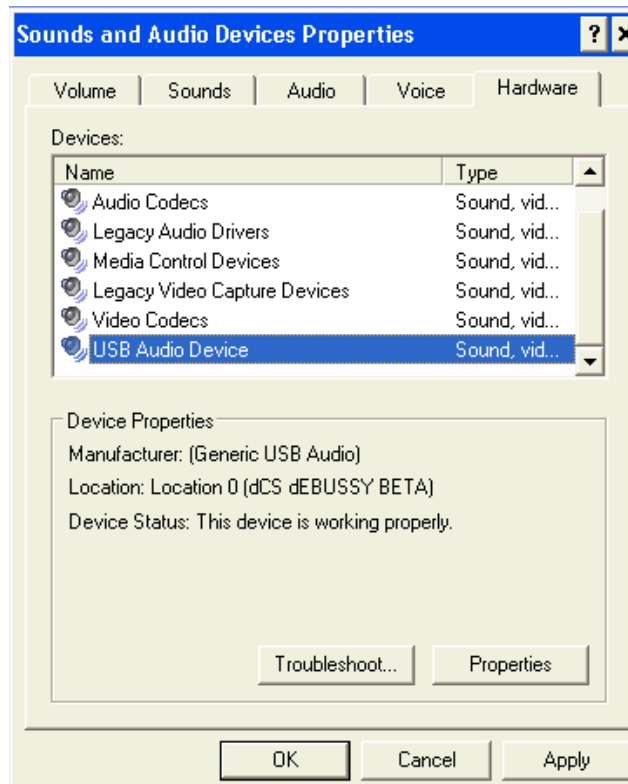
Expand **Audio Devices** by clicking the **+** sign to the left of the speaker icon. Select **USB Audio Device** and click the **Properties** button.
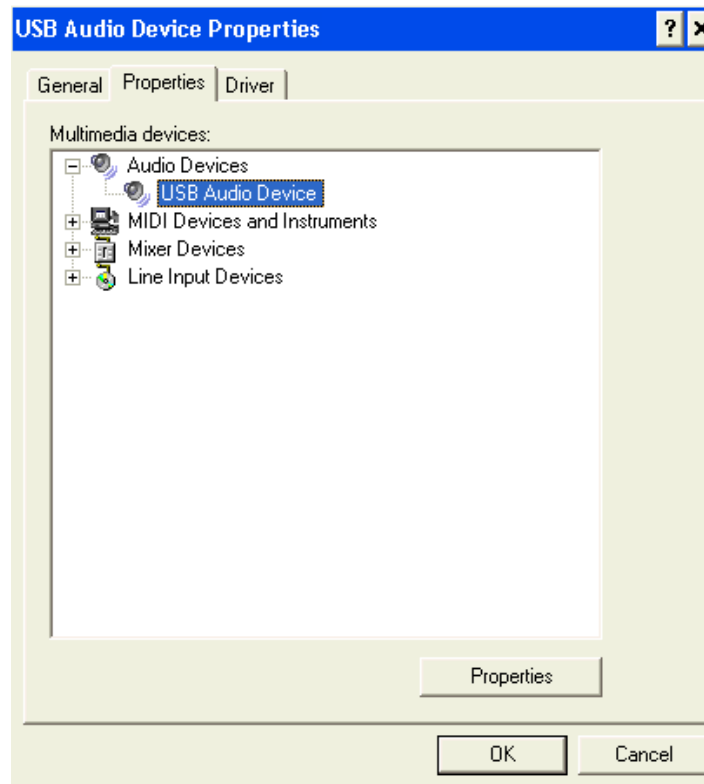
Check the box next to **Do not map through this device**. Re-boot the computer.

Click on **OK** and re-boot the computer.

Windows Vista / Windows 7 Setup

Achieving bit perfect output using Windows Vista or Windows 7 requires a few changes within the Control Panel as well as each playback application. Instructions for unmapping the audio device do not apply to Windows Vista and Windows 7 as this is not an option in newer Microsoft operating systems.

Enter the **Control Panel** and open the **Sound** settings.
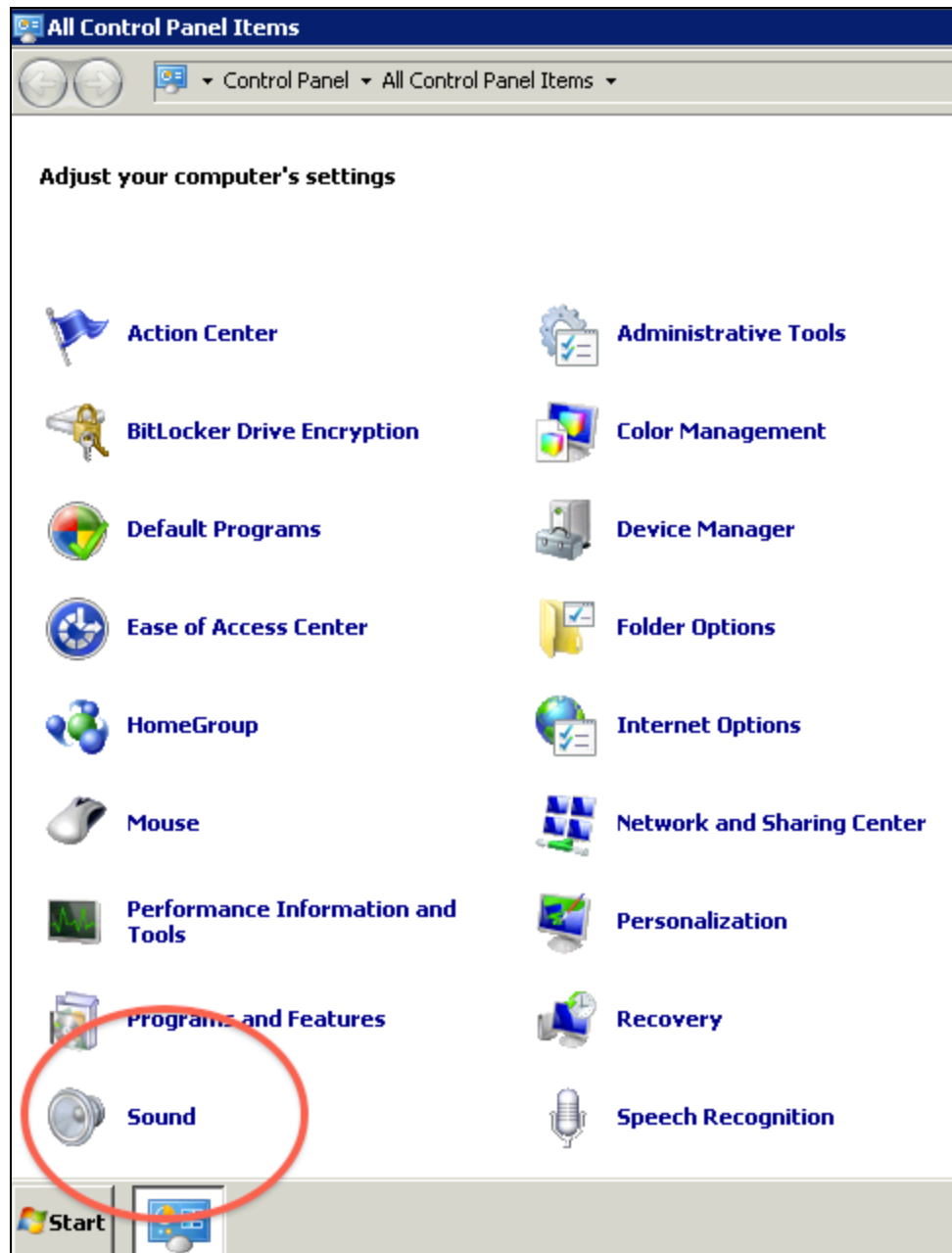
As an extra step to ensure proper configuration, it is recommended an audio device other than the device connected to the audio system be set as the **Default Device**. In the next picture, the **dCS Debussy** is purposely not set as the default device. Setting the laptop or desktop speakers as the default device will give listeners an audible notification something is not configured correctly. If sound is heard from the default computer speakers, reconfiguration is necessary as described. When the default device is set to an internal audio card instead of the computer speakers not connected to the main audio system, no sound will be reproduced if audio is being sent through the operating system instead of directly to the audio card. This is also an indication reconfiguration is necessary.

Right-click on a different device and click on **Set as Default Device**.
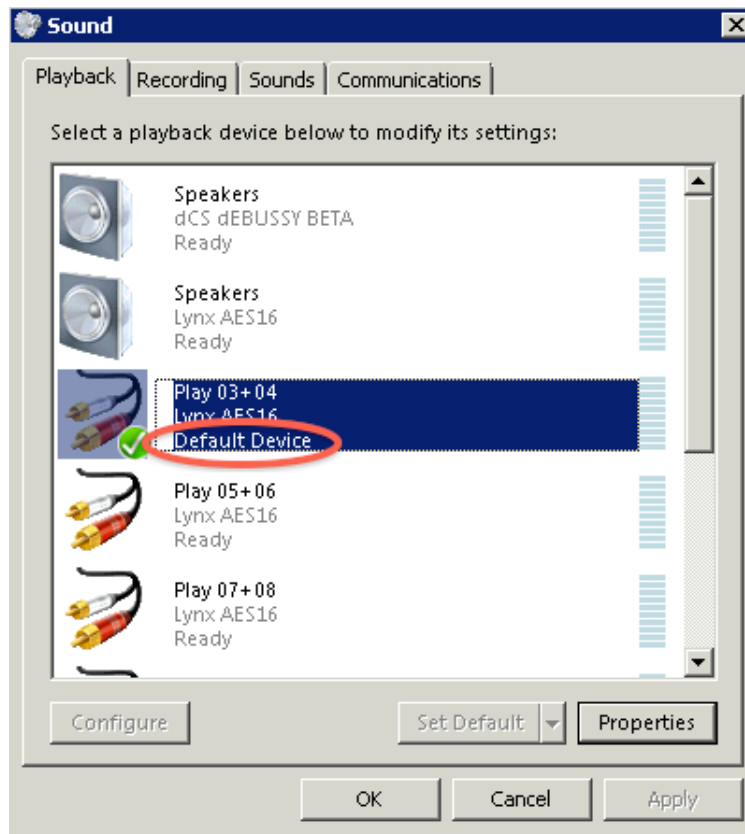
To configure an audio device, select that device then click the **Properties** button.
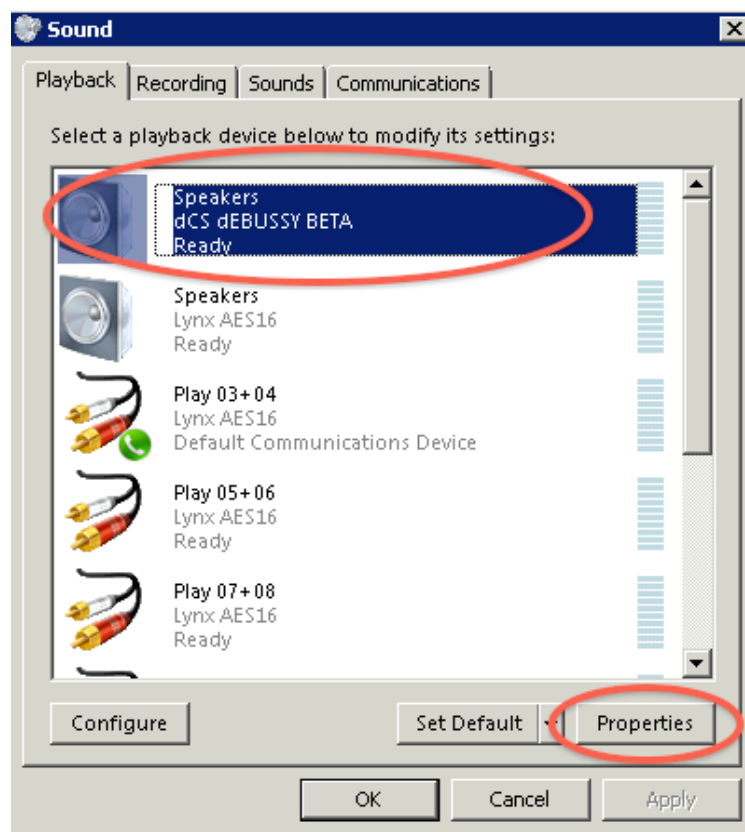
Select the **Enhancements** tab. Select the check box next to **Disable all enhancements**.
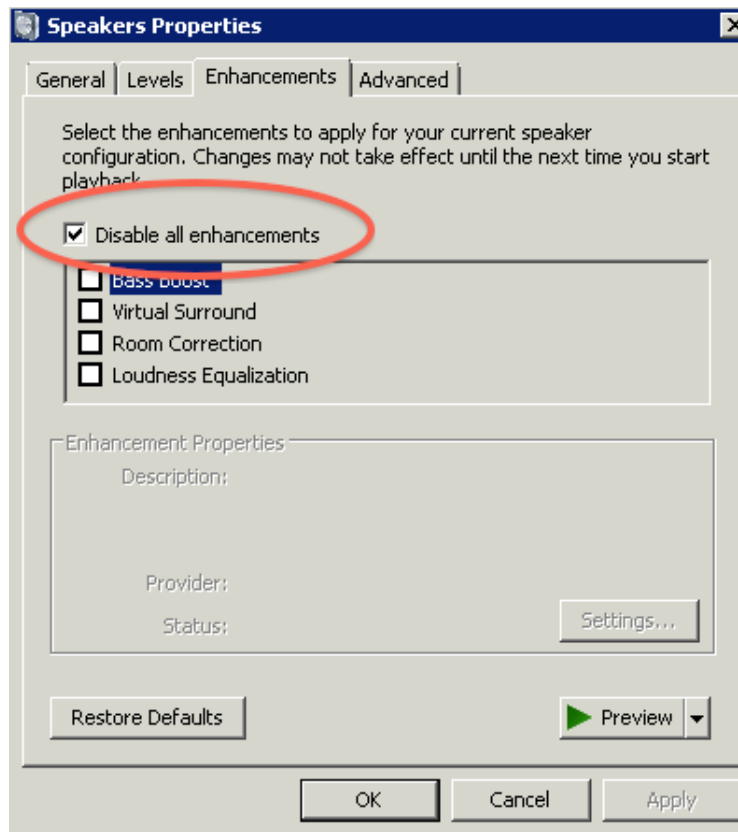
Select the **Advanced** tab and make sure both **Exclusive Mode** options shown below are checked. Under **Default Format,** set the sample rate to **24bit, 96000Hz(Studio Quality)**.
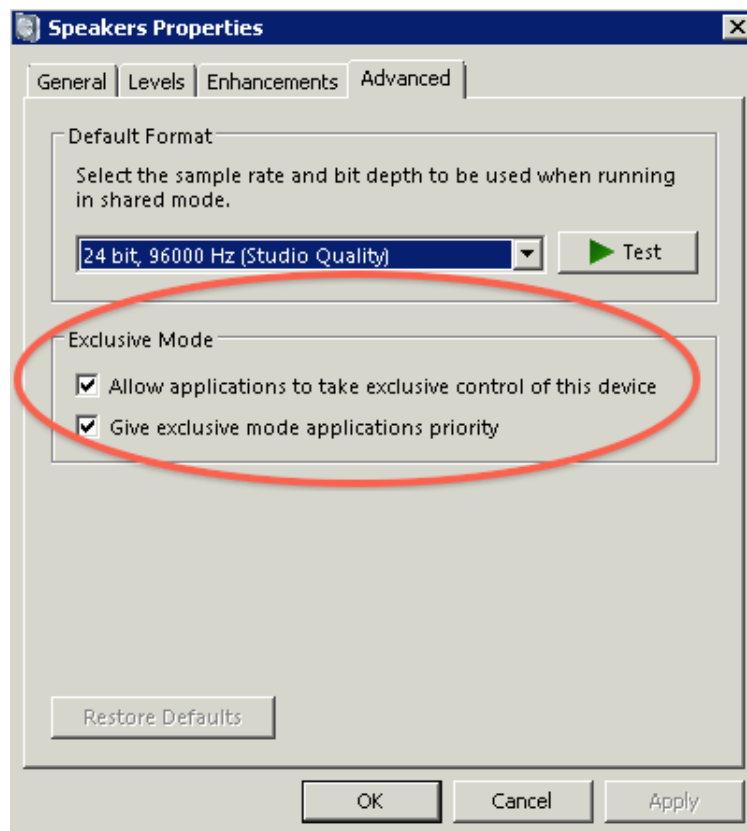
Windows Plugins

There are several software configuration options for outputting digital audio in Windows. The most common Windows plugins / configuration options are listed below, along with a brief explanation of each option.

- **ASIO** - **A**udio **S**tream **I**nput/**O**utput is a common method for sending audio from a playback application directly to an audio device such as a Lynx or RME card. ASIO is a protocol for low latency high quality digital audio. Manufacturers use the ASIO protocol to develop drivers and enable audio card functionality with an ASIO driver. ASIO drivers achieve low latency through bypassing layers of the Windows operating system, such as Windows XP's KMixer. Bypassing KMixer enables bit perfect audio data to reach the audio card. ASIO mode is highly recommended.

- **ASIO4ALL** - Is a universal ASIO driver for Windows that enables many playback applications to send digital audio directly to an audio device such as a USB DAC. ASIO4ALL is most common in consumer audio systems where professional sound cards such as Lynx or RME are not in use.

- **WASAPI** - **W**indows **A**udio **S**ession **API** is similar to ASIO in that it enables audio to be sent directly to a sound device. WASAPI was developed by Microsoft and first appeared in Windows Vista. It remains part of Windows 7. WASAPI by itself is only a tool that software companies can use to enhance audio playback applications. Some applications do not use WASAPI. Currently J River Media Center and Foobar2000 can use WASAPI to send audio directly to the sound device. WASAPI enables software applications such as J River Media Center to take exclusive control over an audio device such as a USB DAC or Lynx / RME audio card. Exclusive control means no audio can be sent to the sound device other than what comes from the playback application. WASAPI is the recommended output mode.

- **DirectSound** and **Wave Out** - these are additional methods for sending digital audio output to a soundcard or device such as a USB DAC. Neither of these methods currently bypass Windows Vista / 7 mixers and multiple layers of the Windows operating system. Severely degraded sound is possible using either DirectSound or Wave Out. That said, it is possible to achieve bit perfect audio while using either method but it's not advised. Higher latency and difficulty maintaining bit perfect playback are two major drawbacks to using DirectSound or Wave Out.

J River Configuration

J River Media Centre 15 has a large number of features that are not relevant to audio performance. The most important options are displayed below.

Download J River Media Centre 15 from www.jrmediacenter.com and install.

Open J River Media Center, select the **Tools** menu then **Options...**
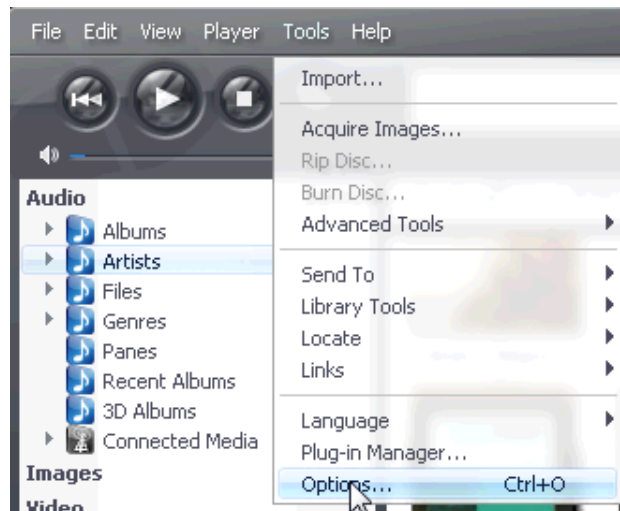
Select **Playback** or **Audio** on the left panel to display the audio output options.
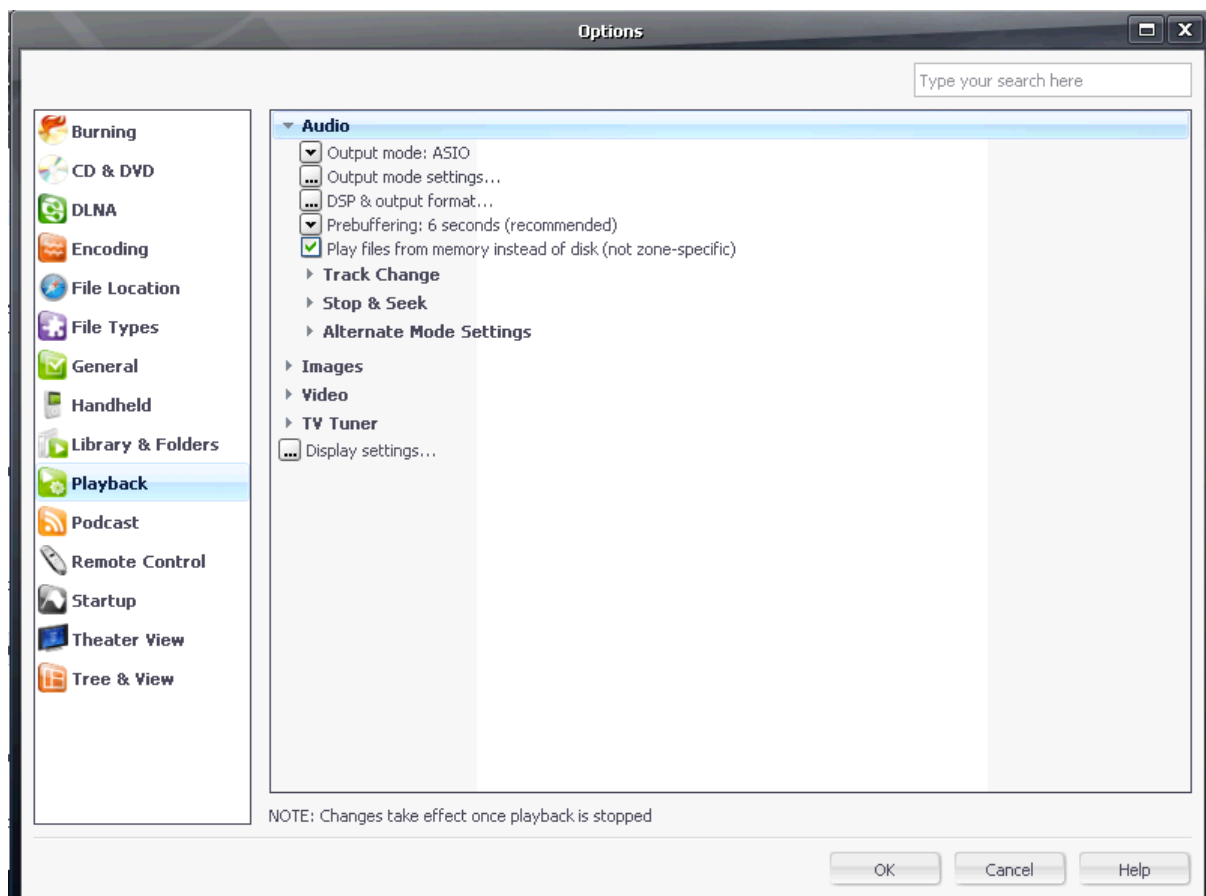
ASIO

The next few instructions describe the use of JRMC in **ASIO** output mode with a USB DAC.

Download ASIO4ALL from www.asio4all.com/ and install.

Select the downward arrow to the left of **Output mode** and select **ASIO**.
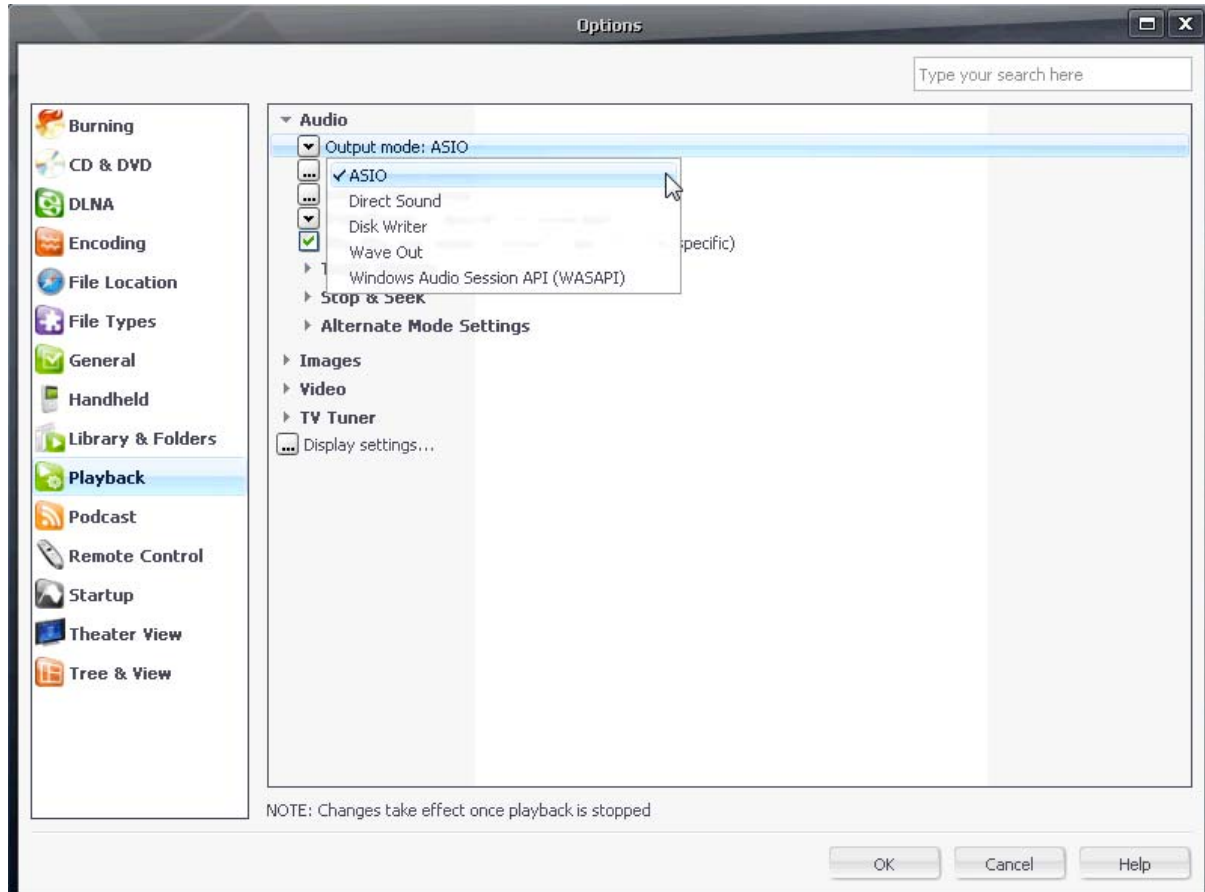
Select the button to the left of **Output mode settings...** to enter the ASIO settings window. Make sure **ASIO4ALL v2** is selected and **Use large hardware buffers** is <u>NOT</u> checked.
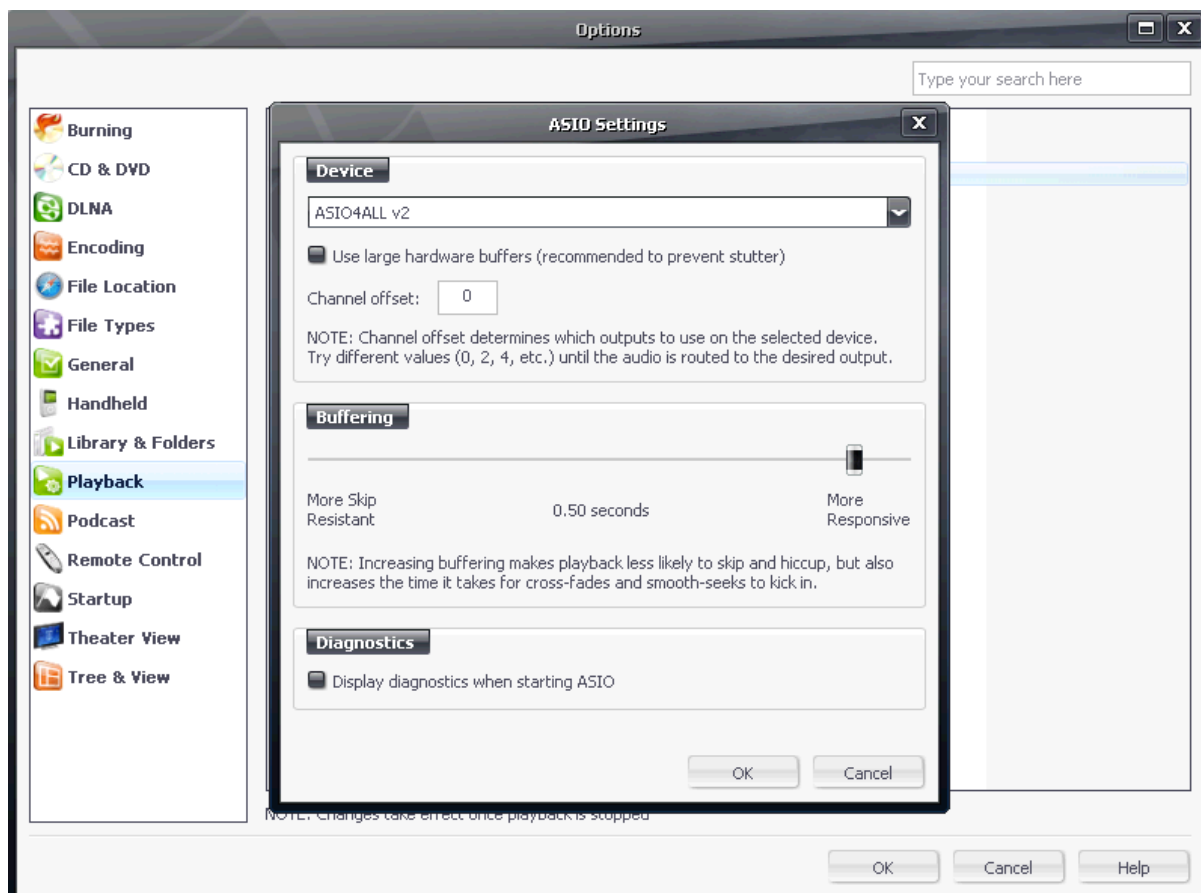
Click **OK** to close the JRMC configuration windows, close JRMC and re-open. To configure ASIO4ALL, start playback of a single track within JRMC. A little green box, as pictured below, will appear in the Windows System Tray. Click the green box to bring up the configuration options.
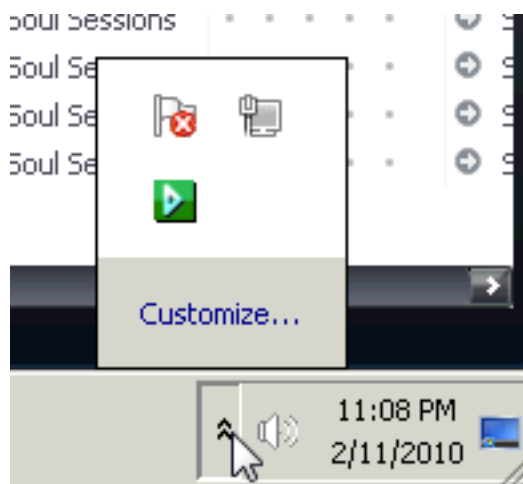
Make sure the USB DAC is listed with an arrow or Play symbol to the left of the DAC label. Simply clicking the device can enable or disable its output. J River Media Center is now configured to output bit perfect audio to a USB DAC.
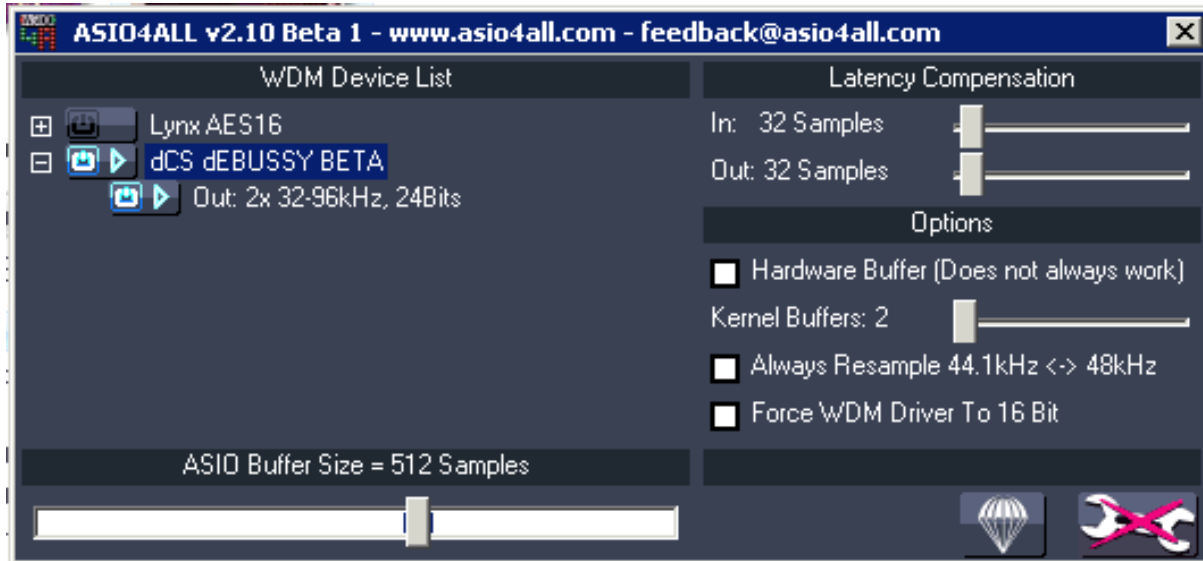
WASAPI

<u>Windows Vista and Windows 7 only</u>: The following instructions describe the use of JRMC in **WASAPI** output mode. WASAPI output mode may not function appropriately with every DAC or audio card because it is a feature of each playback application. If WASAPI is enabled and not working correctly the user may hear ticks and pops during playback, or may not hear any audio signal. Trouble outputting an audio signal is usually visible as the track counter will stop around 0 or 1 second into playback.

Select **Playback** or **Audio** on the left panel to display the audio output options. Select the downward arrow to the left of **Output mode** and select **Windows Audio Session API (WASAPI)**.
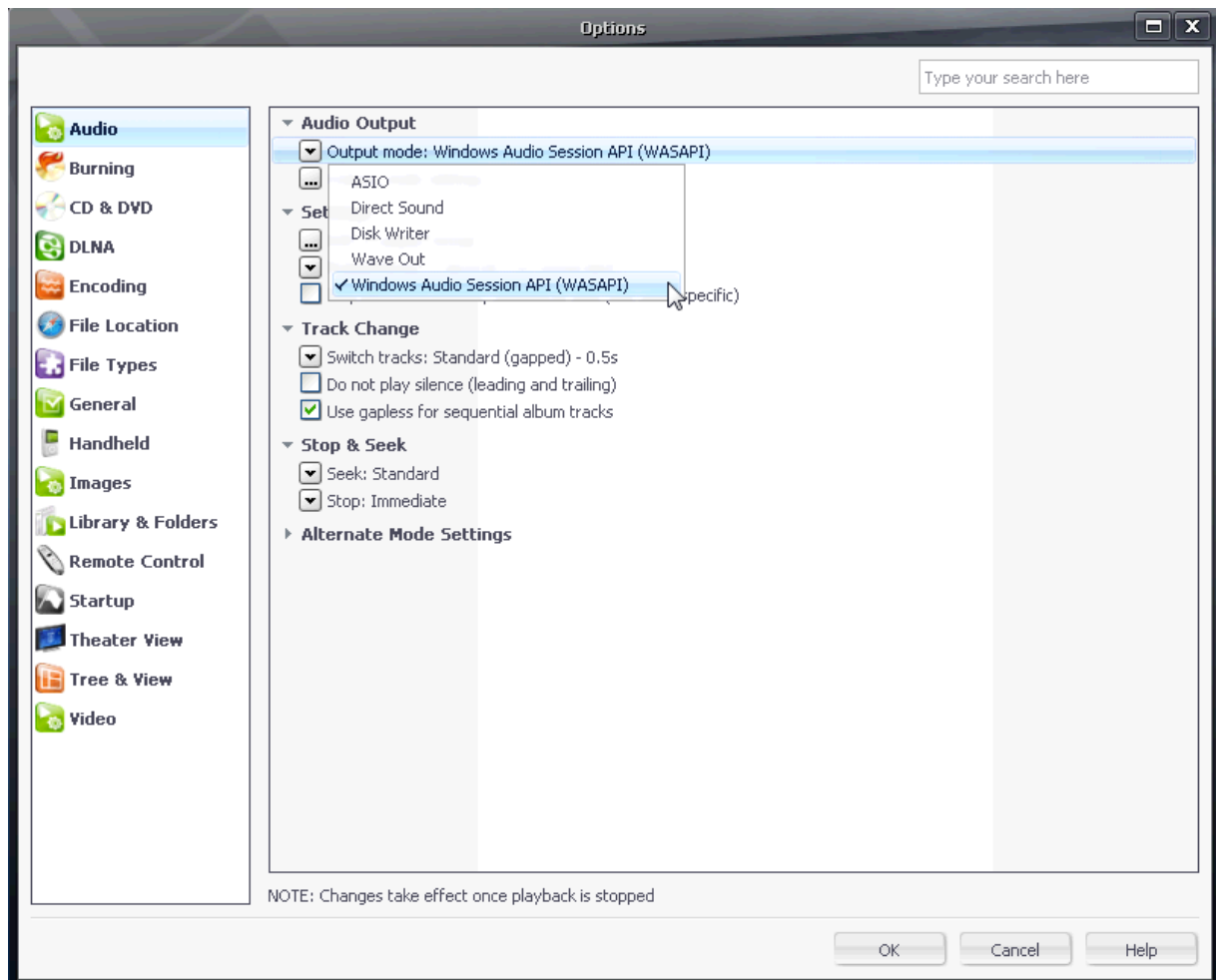
Select the button to the left of **Output mode settings...** to enter the WASAPI settings window. Make sure the correct USB DAC is selected next to **Device** and make sure **Open device for exclusive access** is checked. Set **Buffering** to **0.50 seconds** and click **OK**.
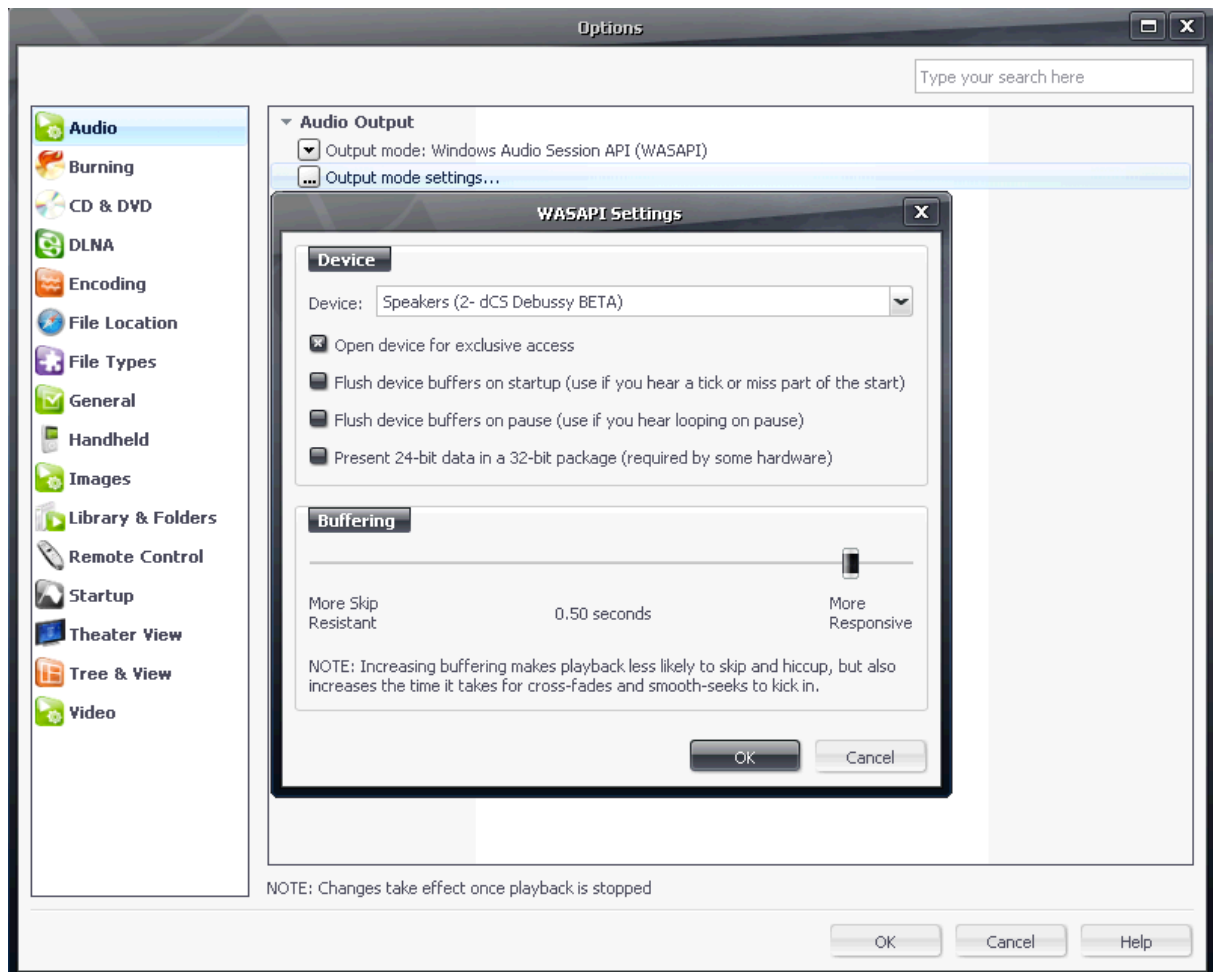
If you hear skipping or stuttering effects, try changing the **Buffering** setting.

Also, try a different PC USB port – the original port may be in use inside the PC.

Select the button to the left of **DSP & output format...** to enter the **DSP Studio** settings window.
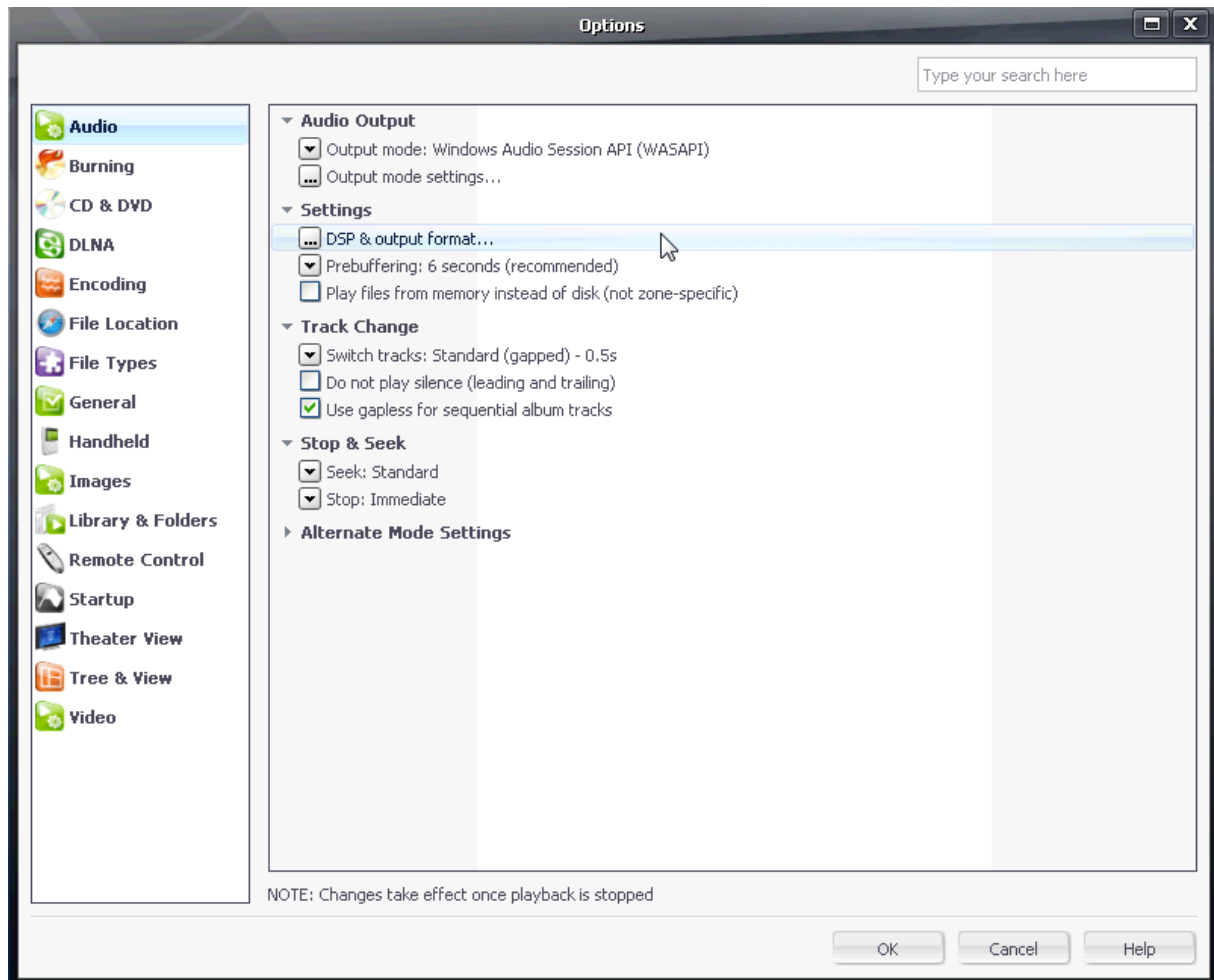
Select **Output Format** on the left pane. Select the drop-down arrow to the right of **Bitdepth:** and select **24-bit**. Click the **X** in the top right corner close the DSP Studio window.
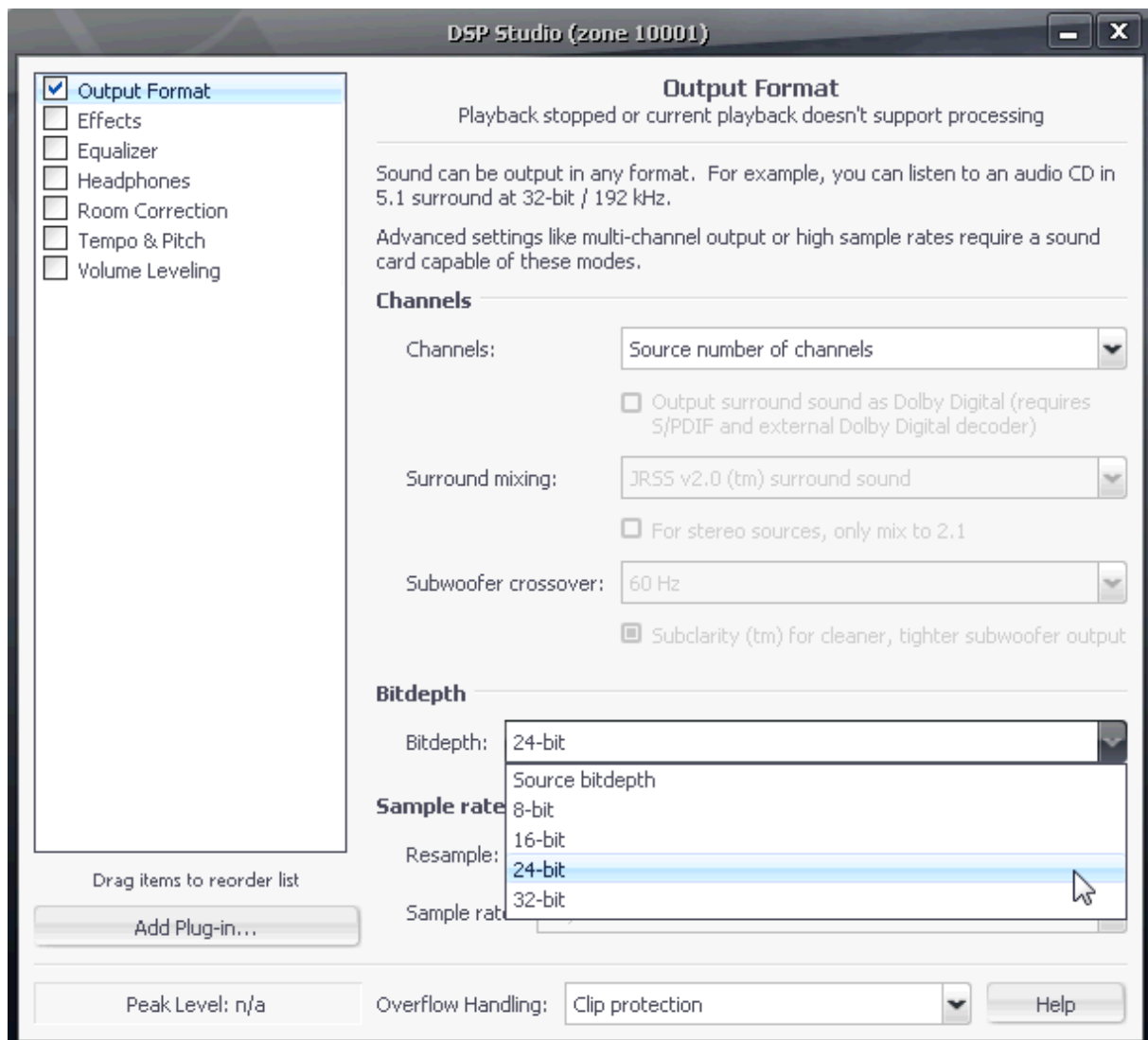
Click **OK** in the audio options window, close JRMC and re-open. WASAPI is now configured.

WASAPI Event Style

Windows Vista and Windows 7 only: A new output mode called WASAPI Event Style has been added to JRiver Media Center v15. This removes an unreliable Windows layer, allowing a more direct path to the driver and hardware.

The following instructions describe the use of JRMC in WASAPI Event Style output mode. WASAPI output mode may not function appropriately with every DAC or audio card because it is a feature of each playback application. If WASAPI is enabled and not working correctly the user may hear ticks and pops during playback, or may not hear any audio signal. Trouble outputting an audio signal is usually visible as the track counter will stop around 0 or 1 second into playback. The following settings have been verified to work with *dCS* USB inputs.

Select **Playback** or **Audio** on the left panel to display the audio output options. Select the downward arrow to the left of **Output mode** and select **WASAPI - Event Style**.
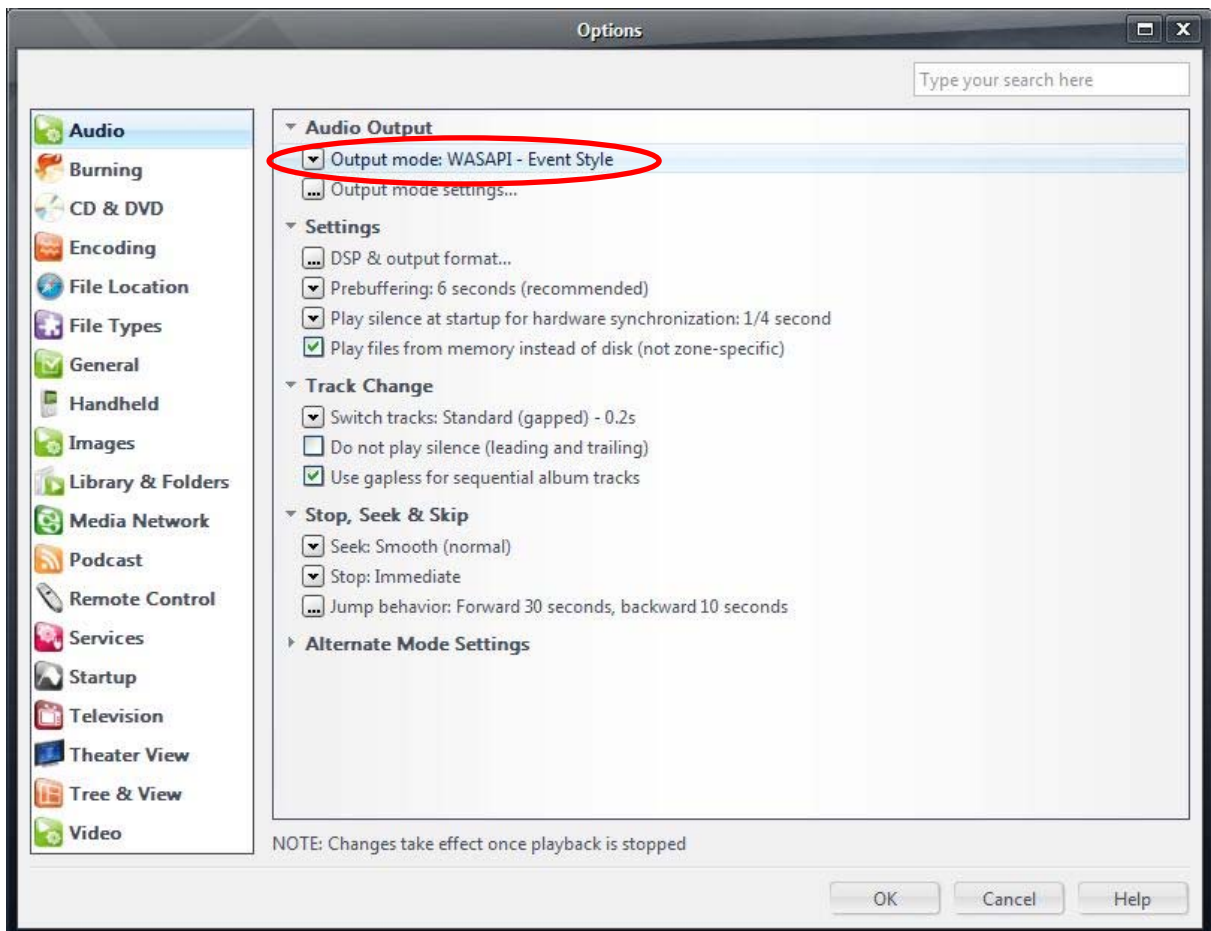
Select the button to the left of **Output mode settings...** to enter the WASAPI settings window. Make sure the correct USB DAC is selected next to **Device** and make sure **Open device for exclusive access** is checked. Set **Buffering** to **100 milliseconds** and click **OK**.
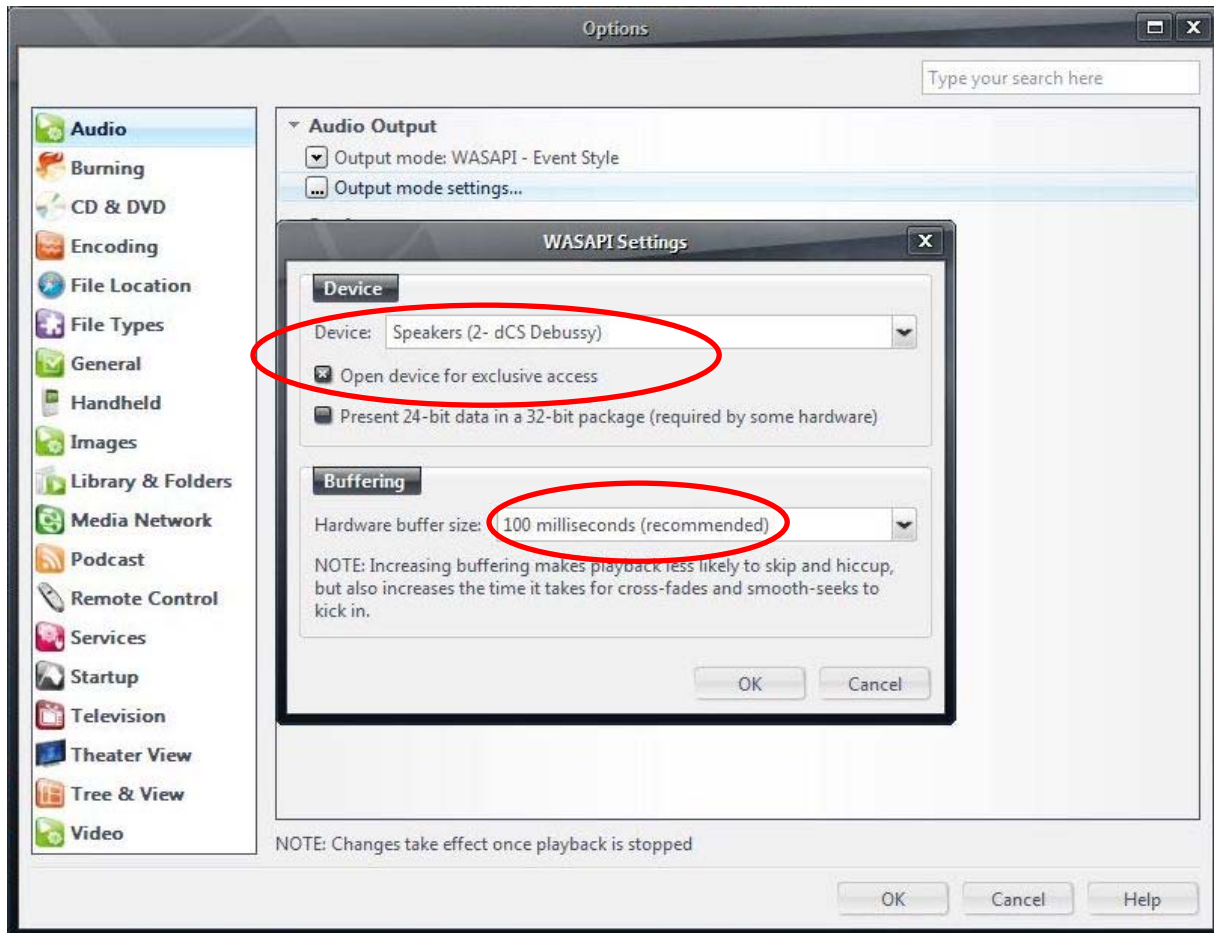
If you hear skipping or stuttering effects, try changing the **Buffering** setting.

Also, try a different PC USB port – the original port may be in use inside the PC.

Select the button to the left of **DSP & output format...** to enter the **DSP Studio** settings window.
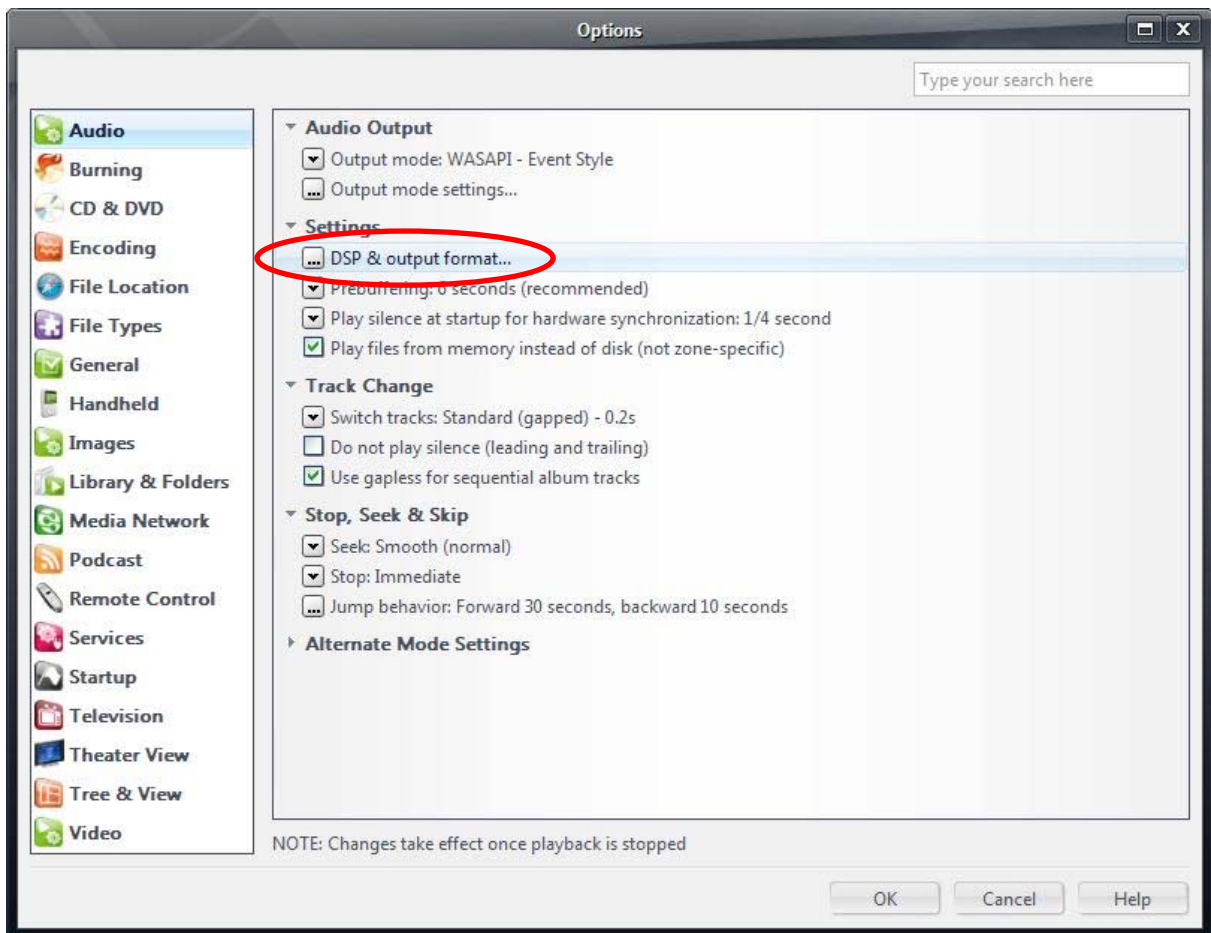
Select **Output Format** on the left pane. Select the drop-down arrow to the right of **Bitdepth:** and select **24-bit**. Click the **X** in the top right corner close the DSP Studio window.
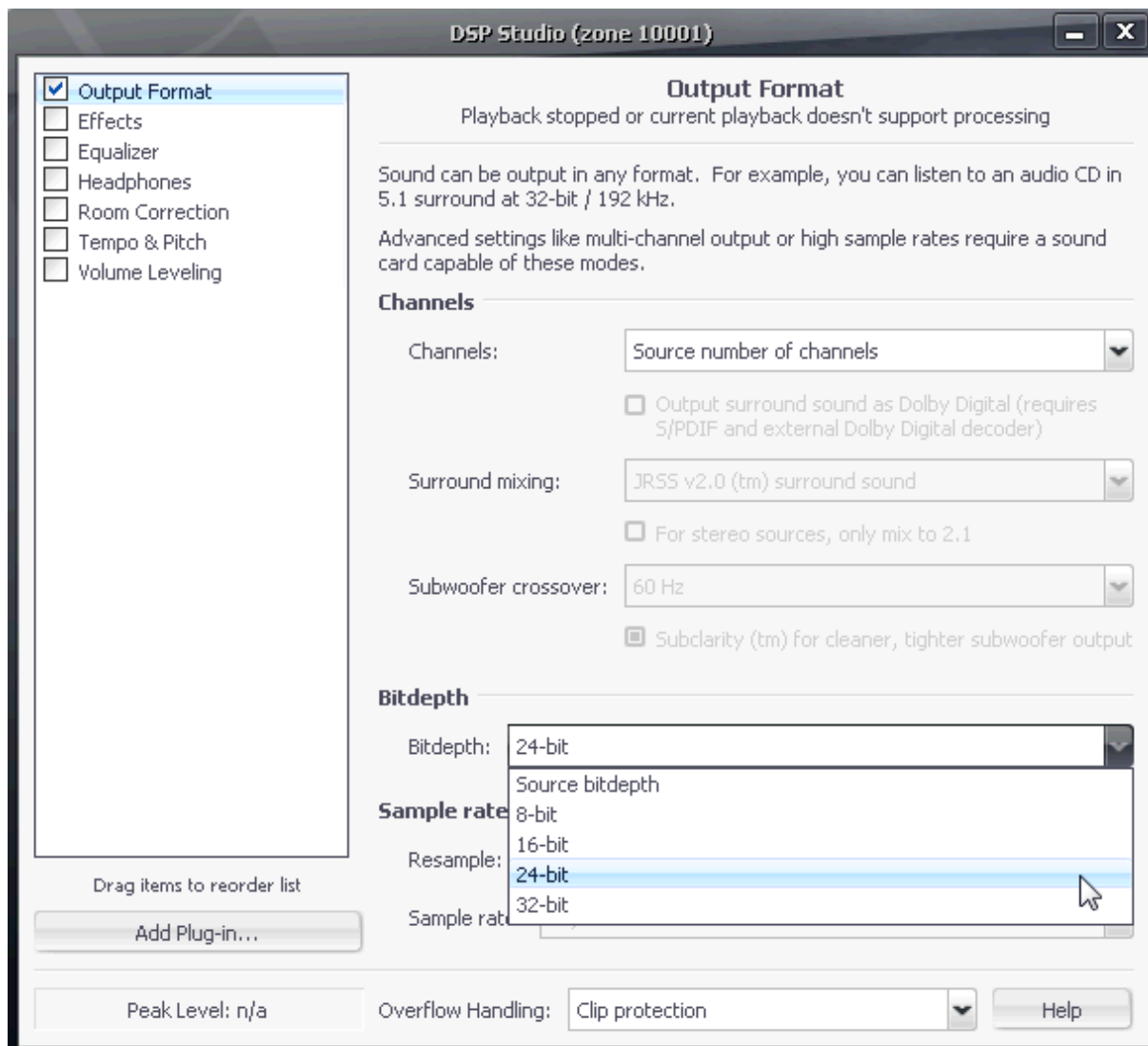
Click **OK** in the audio options window, close JRMC and re-open. WASAPI – Event Style is now configured.

Wave Out

Using the **Wave Out** output mode is not recommended but may be necessary with certain hardware.

Select **Playback** or **Audio** on the left panel to display the audio output options. Select the downward arrow to the left of **Output mode** and select **Wave Out**.
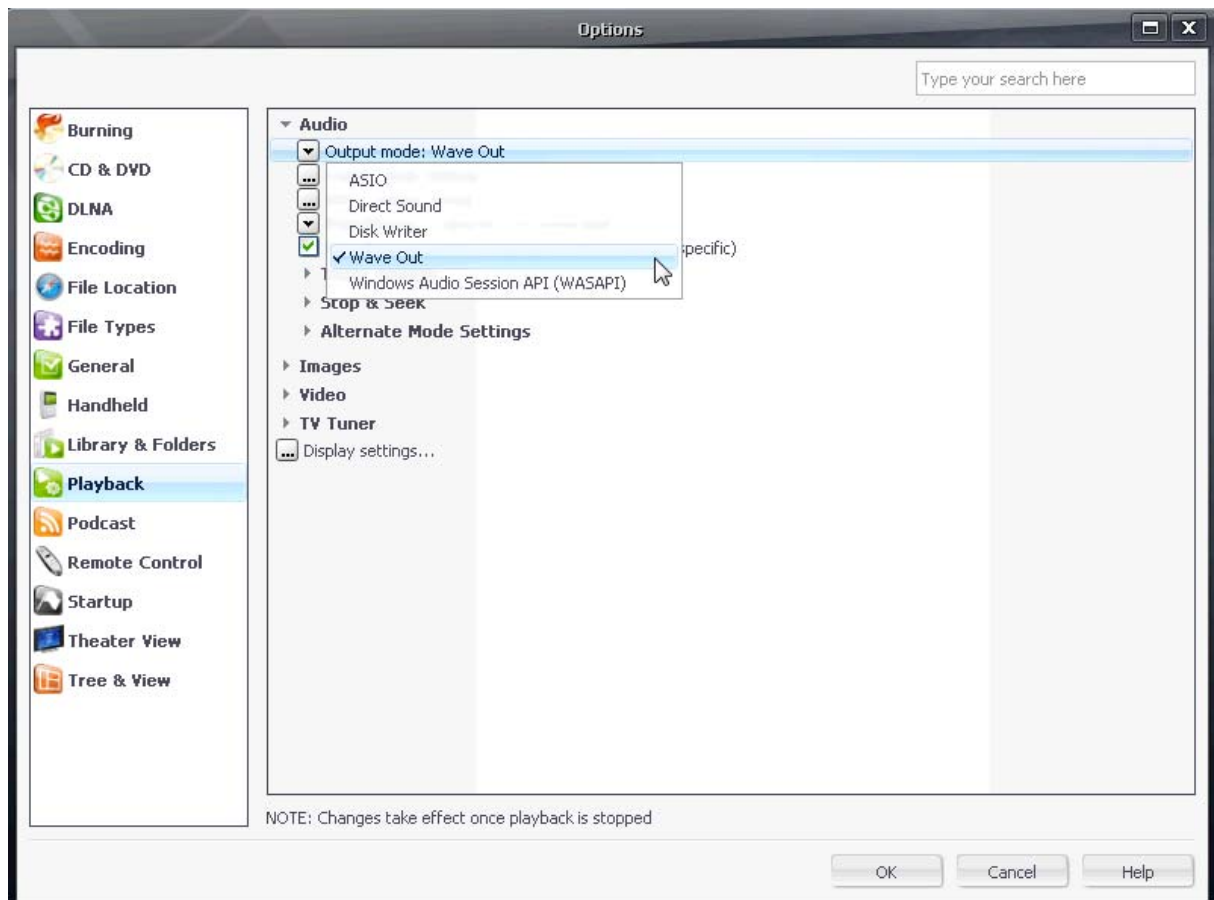
Select the button to the left of **Output mode settings...** to enter the **Wave Out Settings** window. Make sure the correct USB DAC is selected in the **Device** section.
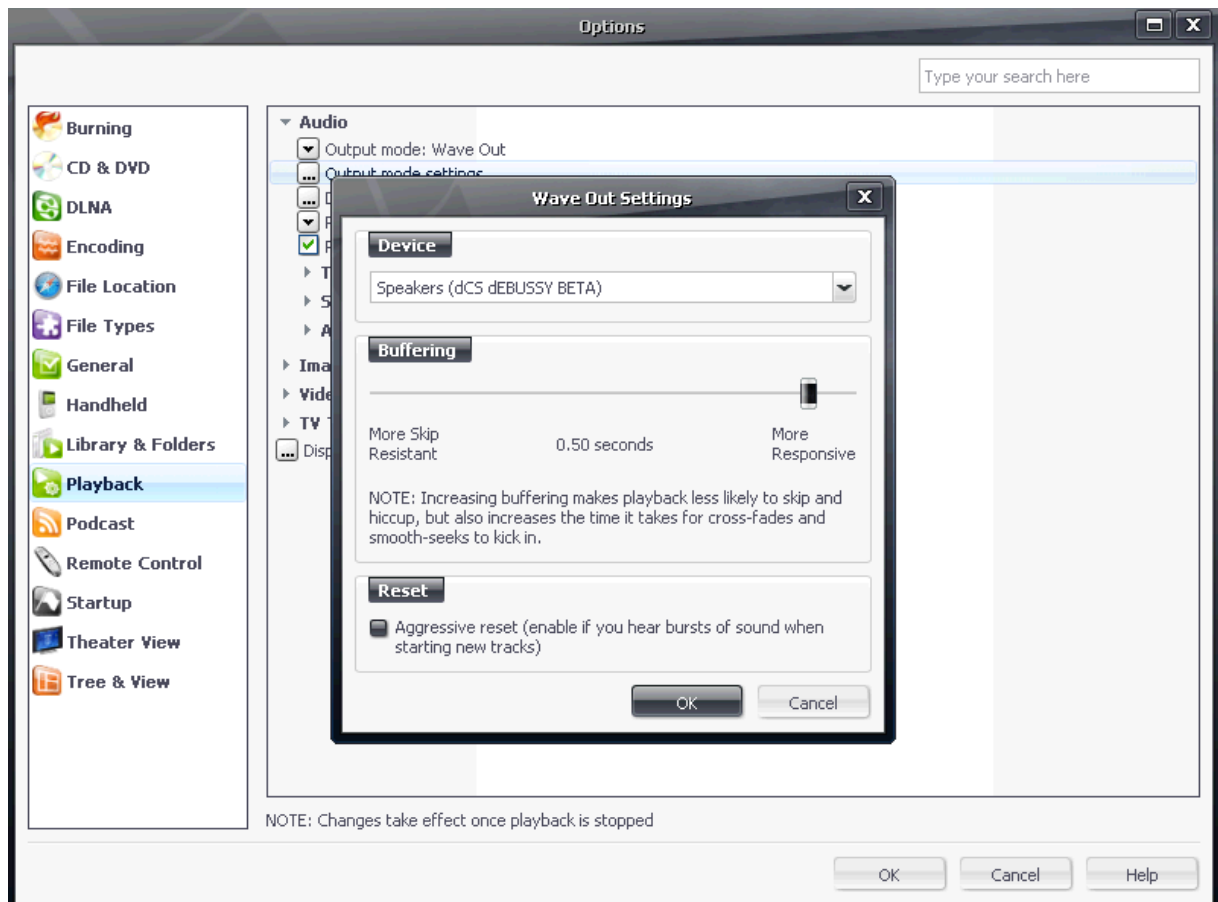
Click **OK**, close JRMC and re-open to conclude the Wave Out configuration.

Foobar Configuration

Download Foobar2000 from www.foobar2000.org/ and install. The recommended output options are described below.

ASIO

The following instructions describe the use of Foobar2000 in ASIO output mode.

Download ASIO4ALL from www.asio4all.com/ and install.

Download ASIO Support from www.foobar200.org/components/. Unzip the file foo_out_asio.dll and drag it to the directory C:\Program Files\foobar2000\components.

Open Foobar, select the **File** menu then **Preferences**.
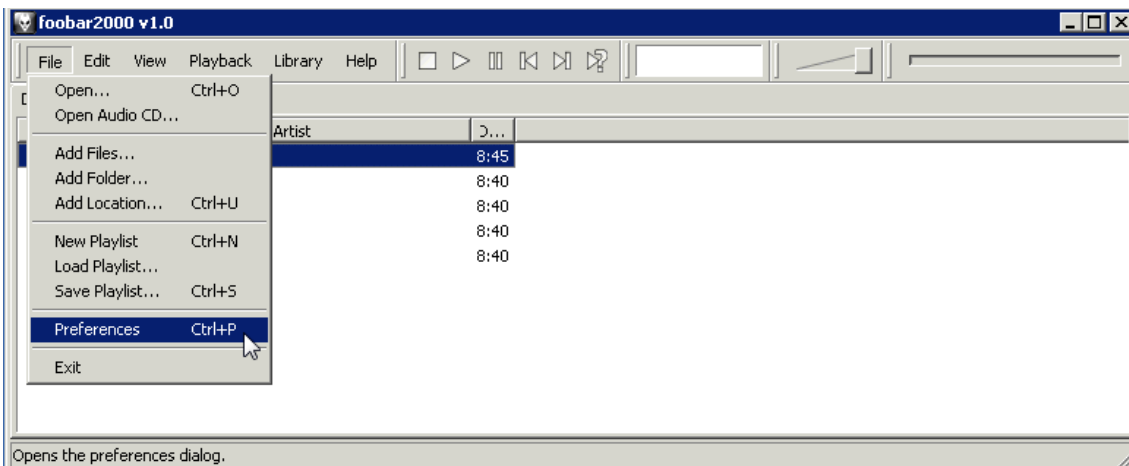
Select **Playback** > **Output** on the left menu tree to display current output settings.

Make sure **ASIO : ASIO4ALL v2** is selected as the **Output Device**.
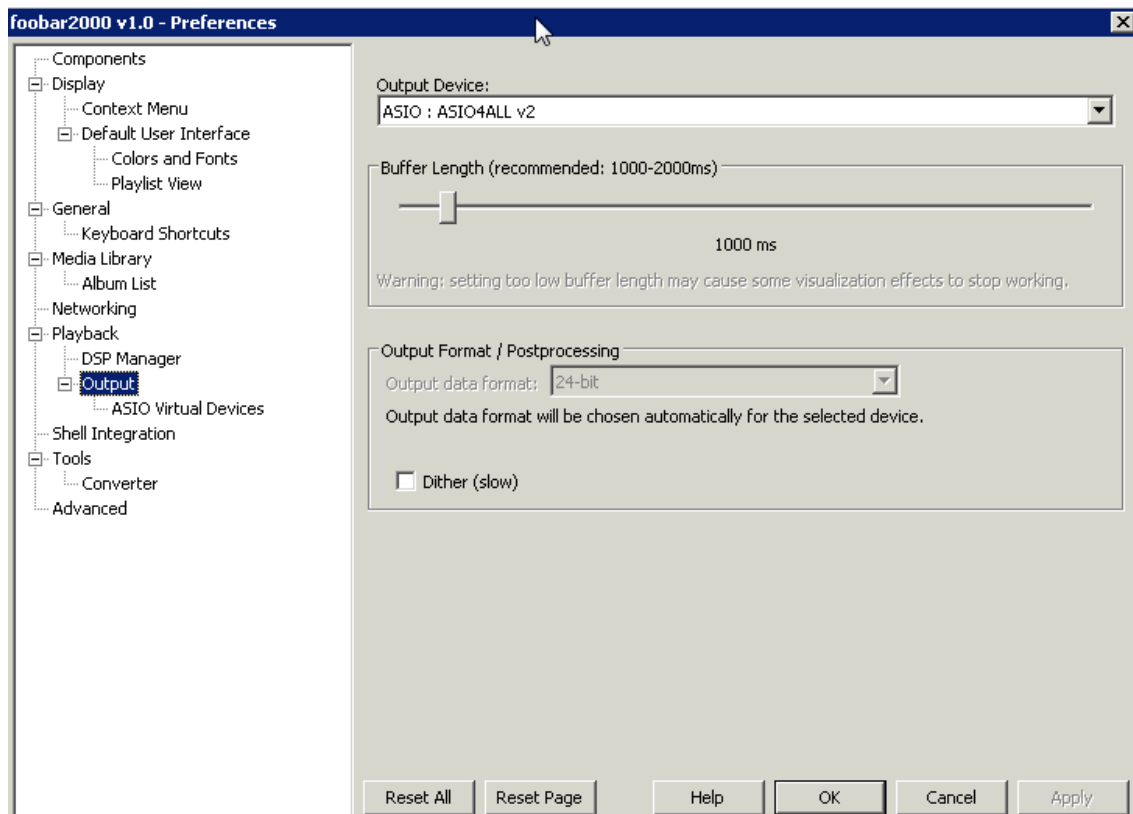
Select **ASIO Virtual Devices** on the left menu tree then click the **Add New** button.
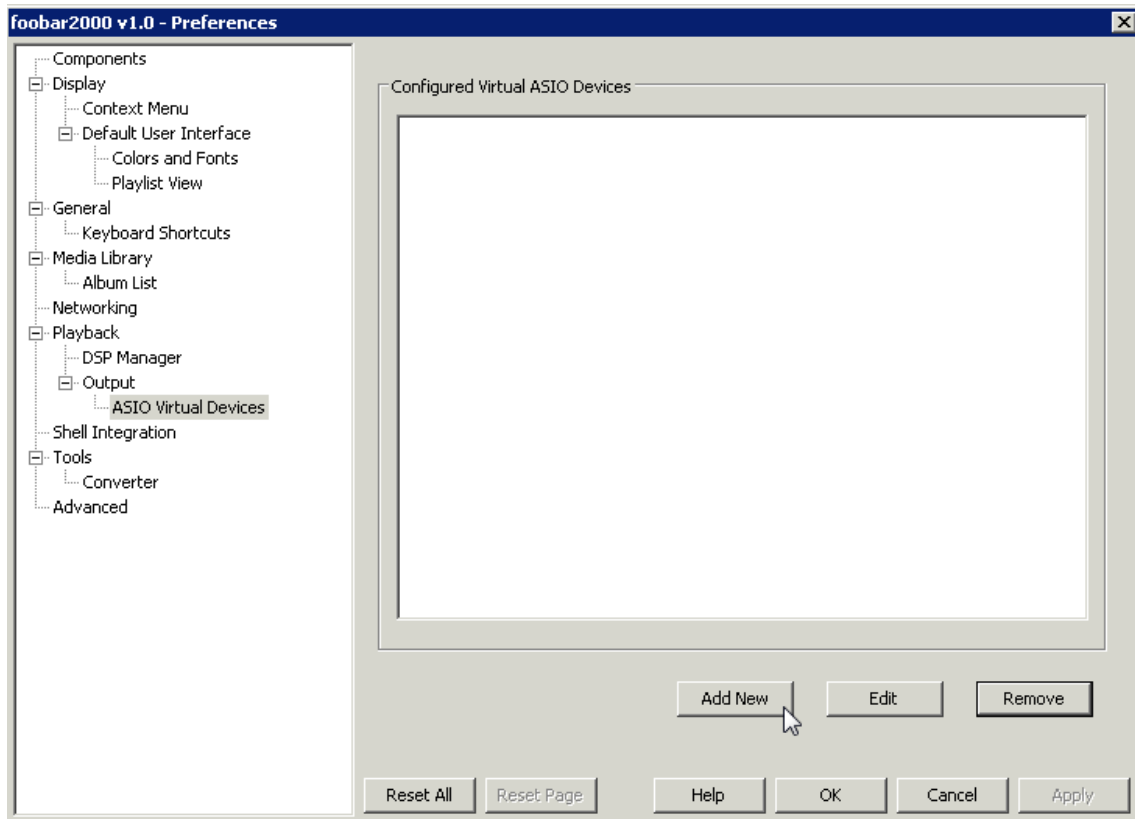
On the **ASIO Virtual Device Editor** window make sure the **Driver** selected is **ASIO4ALL v2**. A single USB DAC will be listed on two separate lines in the **Channel Map** window.
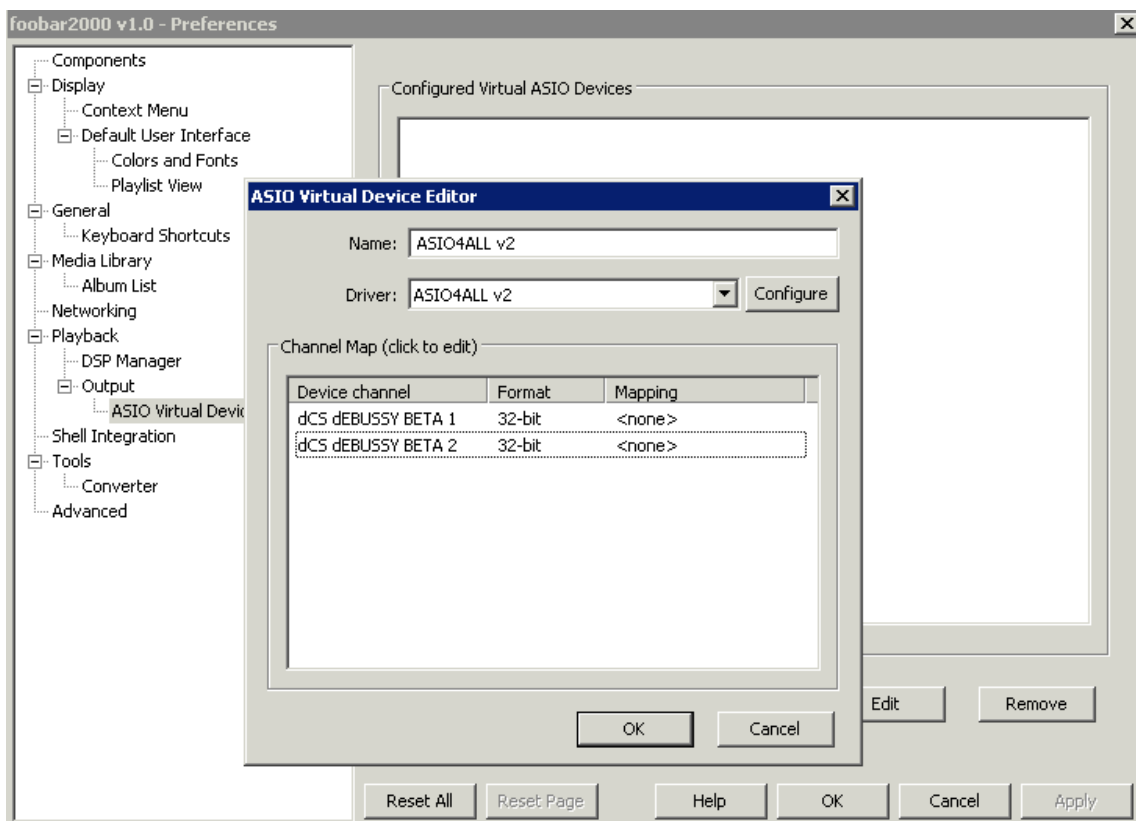
Each line displaying the USB DAC should be mapped to a single output channel. This is done by clicking each line separately and selecting the correct channel. This configuration can be changed via the same procedure if the channels are reversed.
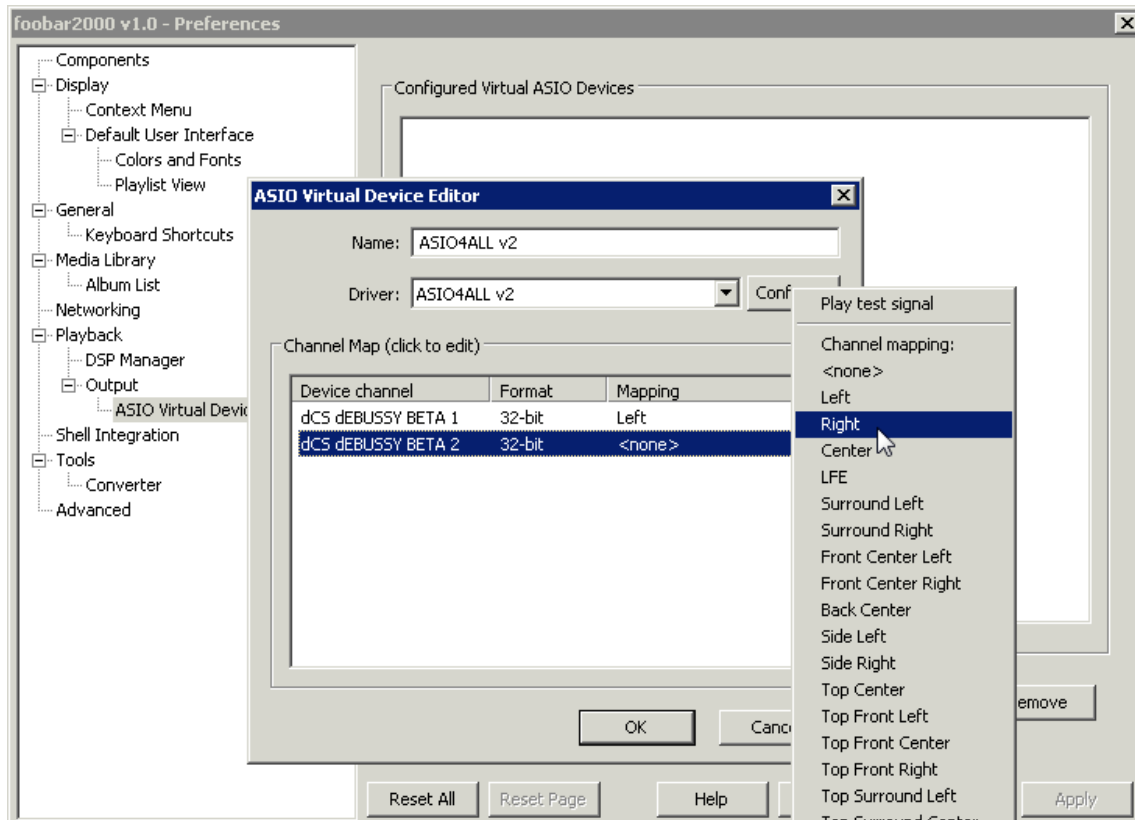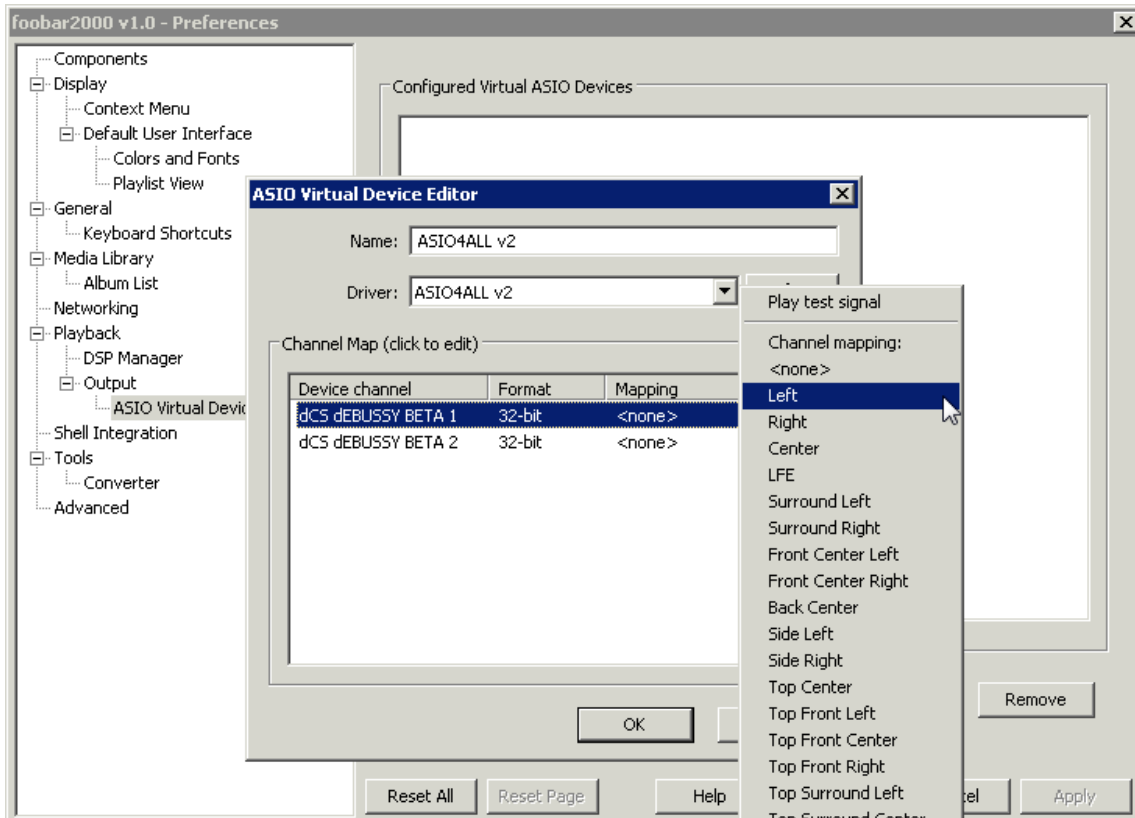
Once the channels are mapped click **OK** to display the single configured virtual ASIO device. It should show ASIO4ALL v2 (unless the name was changed in the previous section).
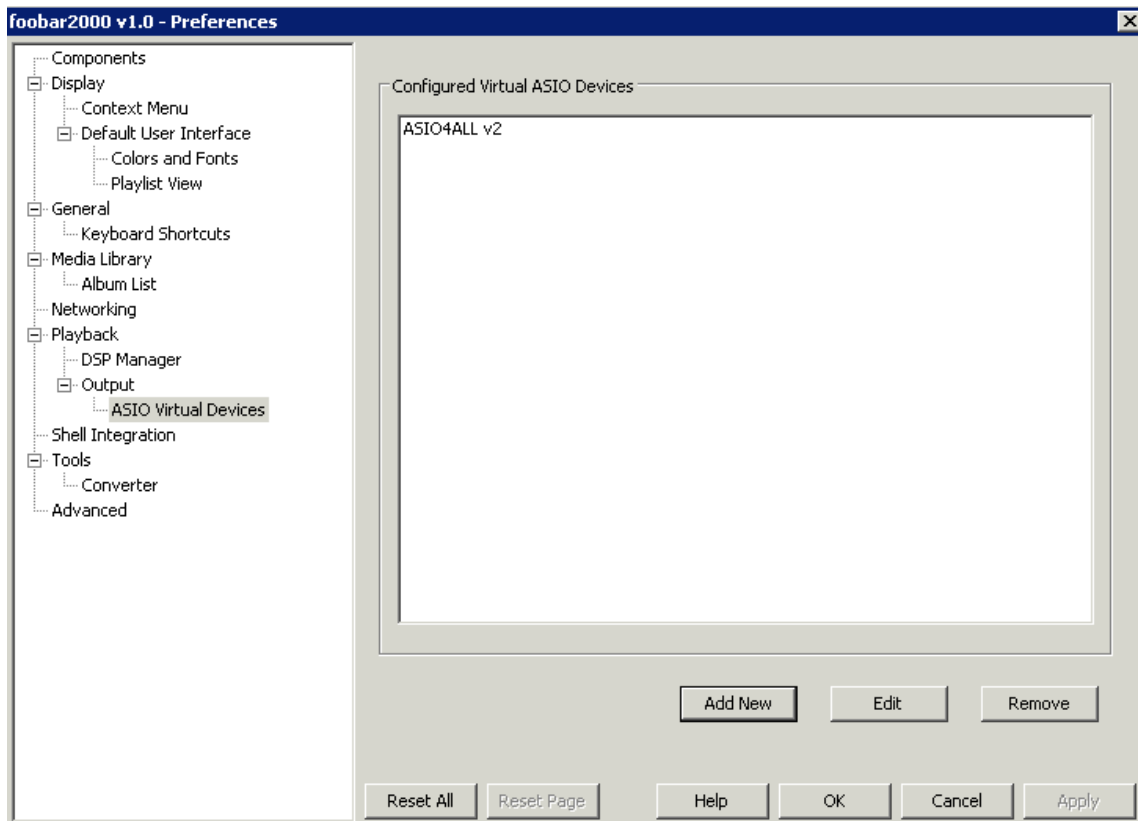
Click on **OK**, close Foobar and then re-open. To complete configuration of ASIO4ALL start playback of a single track within Foobar. A little green box, as pictured below, will appear in the Windows System Tray. Click the green box to bring up the configuration options.
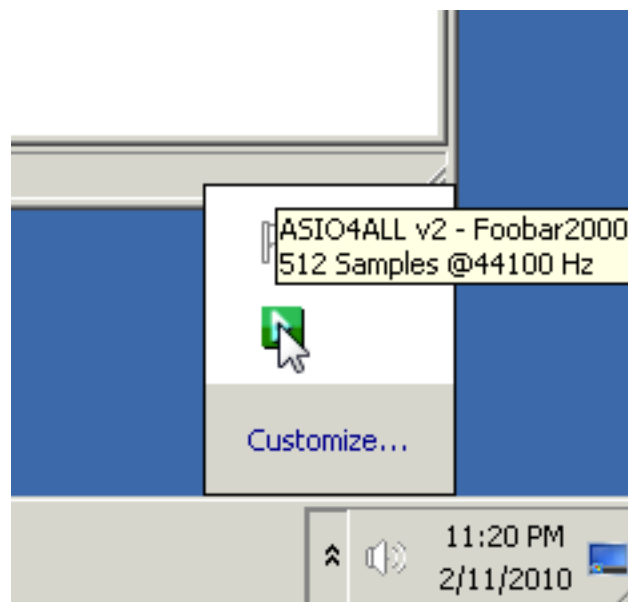
Make sure the USB DAC is listed with an arrow or Play symbol to the left of the DAC label. Simply clicking the device can enable or disable its output. Foobar is now configured to output bit perfect audio to a USB DAC.
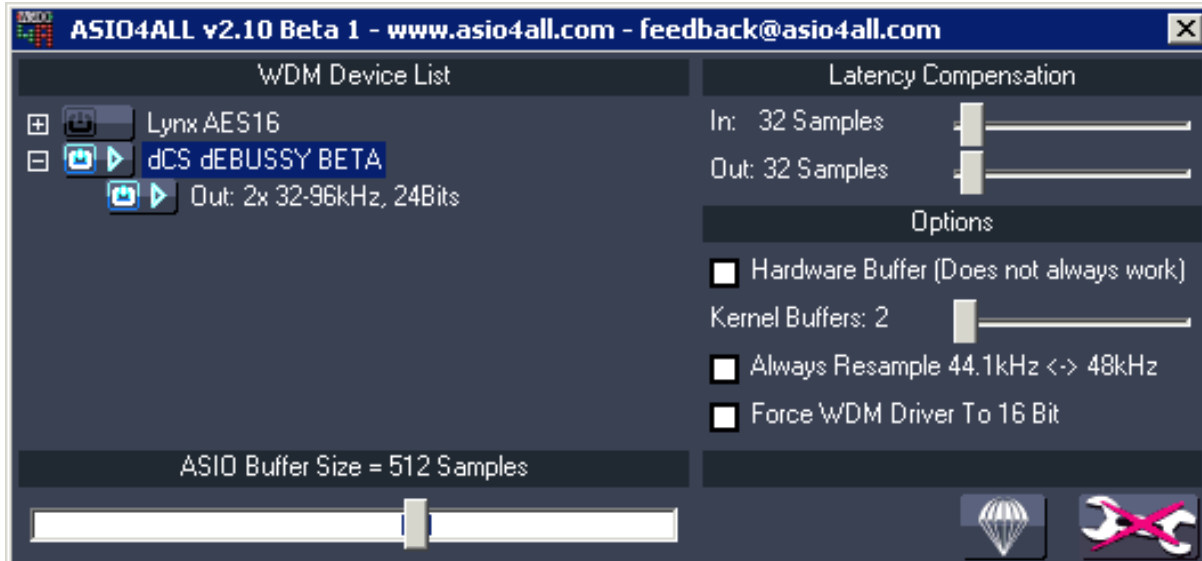
WASAPI

<u>Windows Vista and Windows 7 only</u>: The following instructions describe the use of Foobar2000 using WASAPI output mode.

Open Foobar, select the **File** menu then **Preferences**.
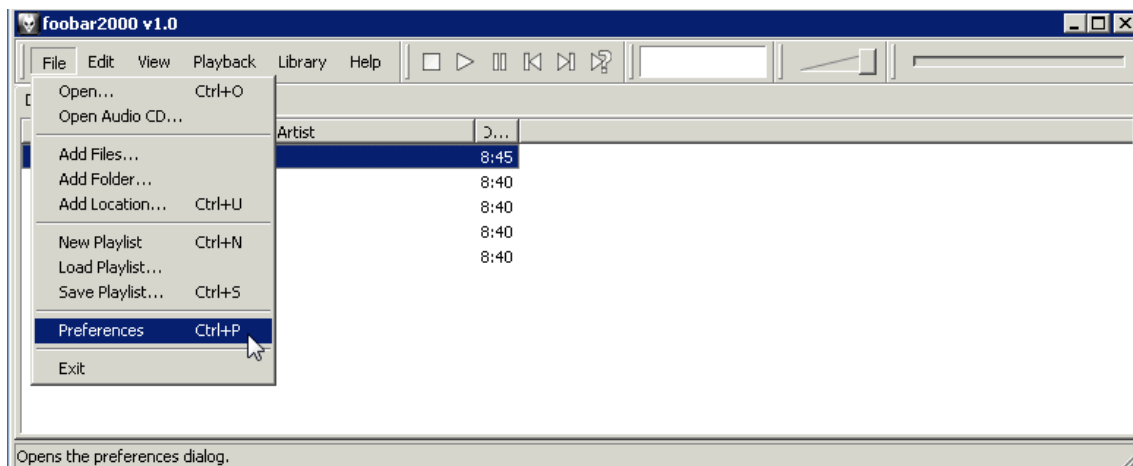
Select **Playback** > **Output** on the left menu tree to display current output settings.

In the right-hand pane, set the **Output Device** to WASAPI for the USB DAC (**WASAPI: Speakers (dCS Debussy BETA4)** in this example). Set the **Buffer Length** to **1000ms** and the **Output Data Format** to **24-bit**.
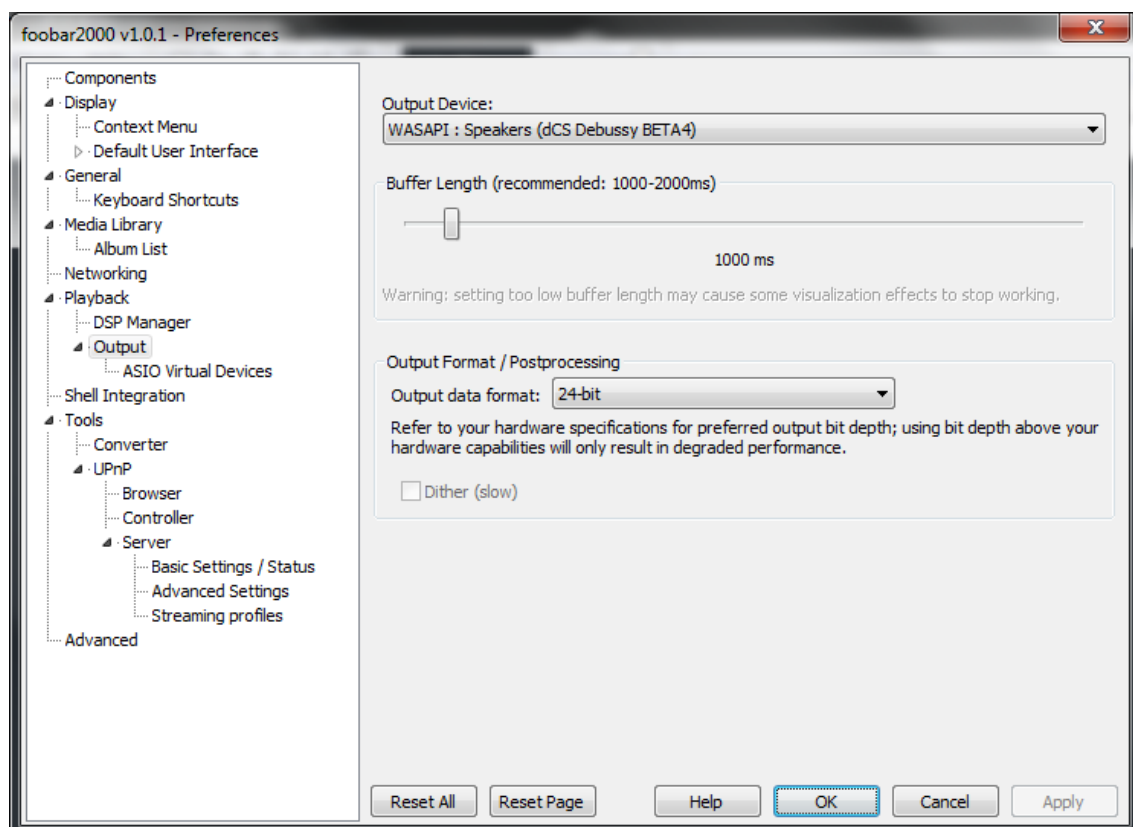
⚠️ If you hear skipping or stuttering effects, try changing the **Buffering** setting.

Click on **OK**, close Foobar and then re-open. WASAPI is now configured.

MediaMonkey Configuration

MediaMonkey v4 (being beta tested at the time of writing), will support WASAPI when it is released. MediaMonkey v3 does not support WASAPI playback. The following instructions describe configuration using ASIO and Wave Out.

Download MediaMonkey from **www.mediamonkey.com/download/** and install.

ASIO

The following instructions describe the use of MediaMonkey in ASIO output mode.

Download ASIO4ALL from **http://www.asio4all.com/** and install.

Download the MediaMoney ASIO plugin from **http://otachan.com/out_asio(dll)_067.7z.**

This plugin download will require extraction from the 7z format. A common tool to extract the needed files can be downloaded from **http://www.7-zip.org/**.

Once the downloaded files are extracted, drag the file out_asio(dll).dll **from** the temporary directory ...\out_asio(dll)_067\bin\SSE2\ **to** C:\Program Files\MediaMonkey\Plugins\ .

Once ASIO4ALL, the MediaMonkey ASIO plugin, and the MediaMonkey application are installed, open MediaMonkey. Select the **Tools** menu and click **Options**.
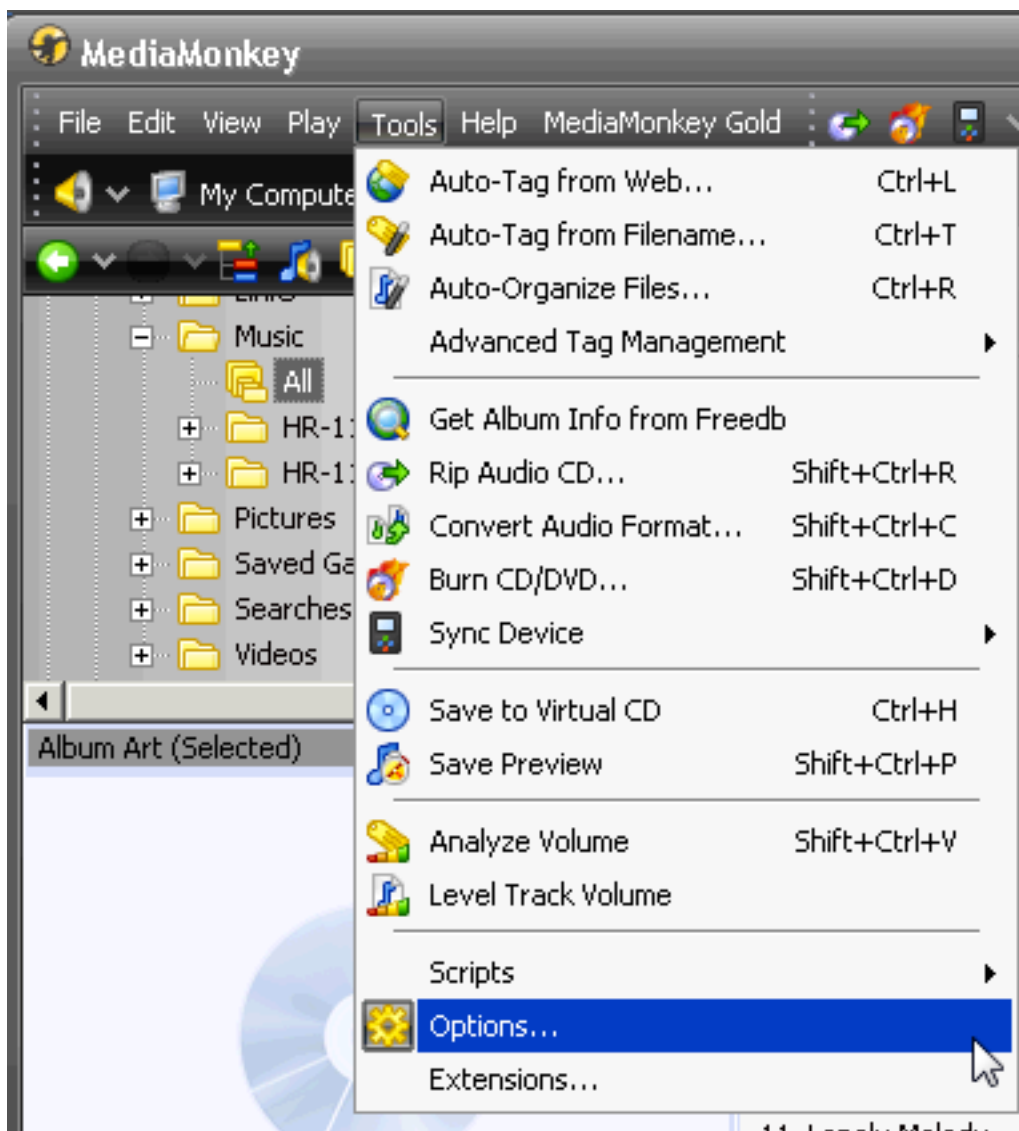
Select **Volume Leveling** from the left panel. Make sure <u>none</u> of the options on the right are checked.
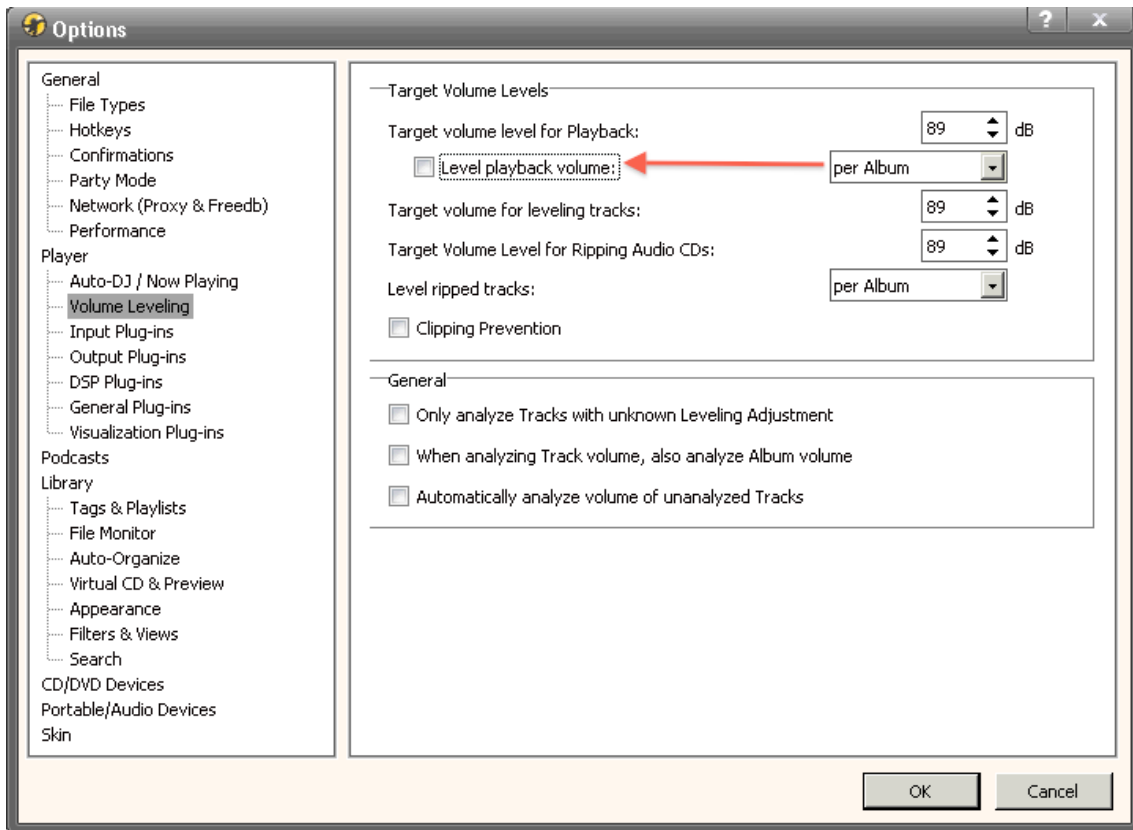
Select **Output Plug-ins** on the left and **ASIO output** on the right. Then click the **Configure** button to enter the ASIO output settings.
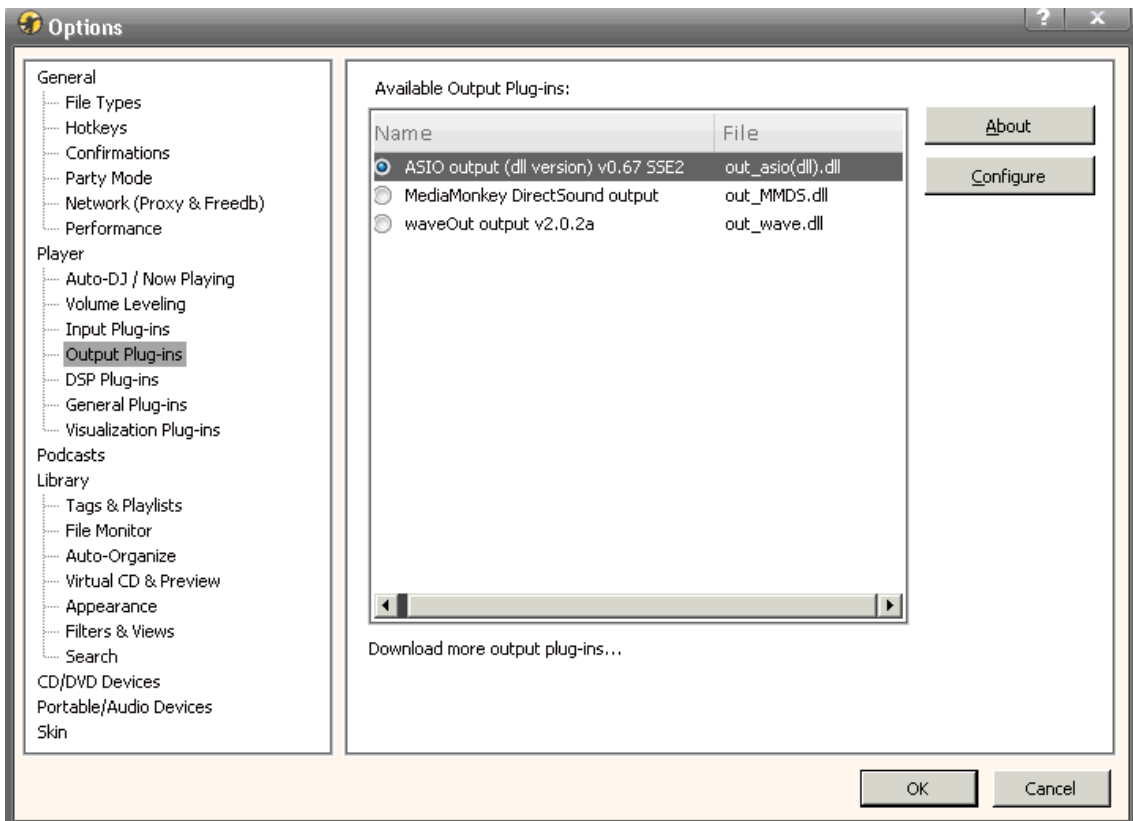
Make sure **ASIO4ALL v2** is selected to the right of **Device** and <u>none</u> of the other checkbox items are checked.
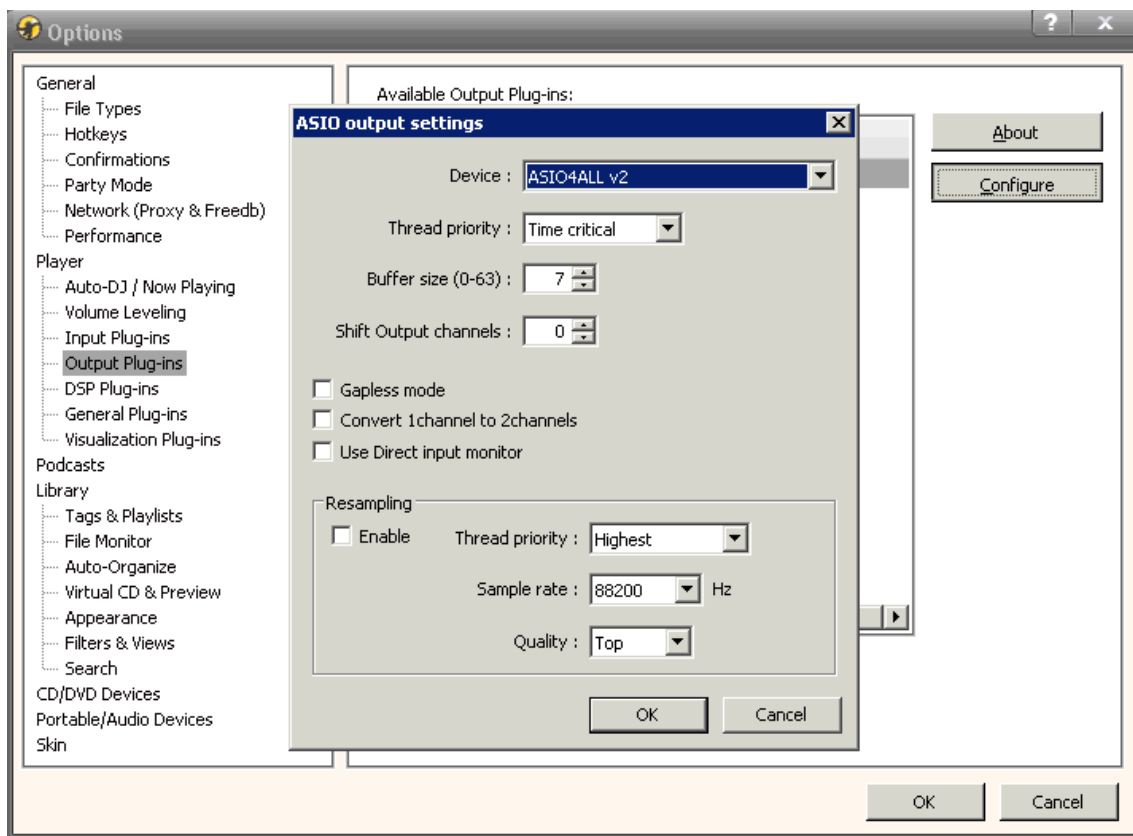
Click on **OK**, close MediaMonkey and re-open. To complete configuration of ASIO4ALL, start playback of a single track within MediaMonkey. A little green box, as pictured below, will appear in the Windows System Tray. Click the green box to bring up the configuration options.
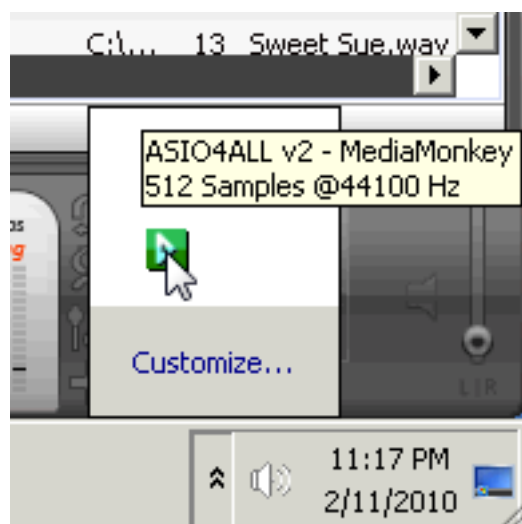
Make sure the USB DAC is listed with an arrow or Play symbol to the left of the DAC label. Simply clicking the device can enable or disable its output. MediaMonkey is now configured to output bit perfect audio to a USB DAC.
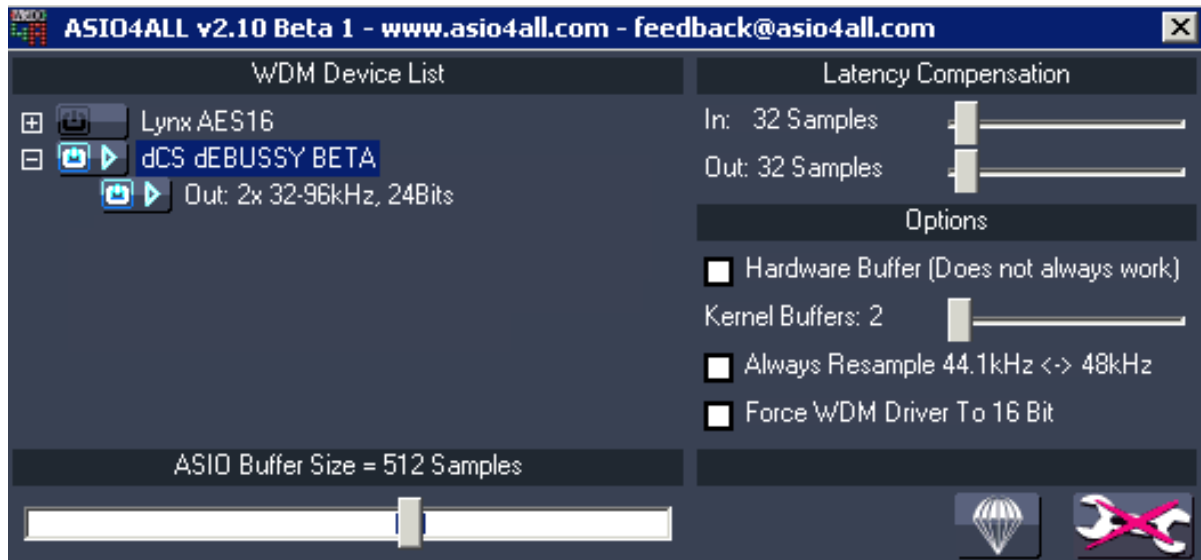
Wave Out

Using the WaveOut plugin is not recommended but may be necessary with certain hardware.

Open MediaMonkey, select the **Tools** menu and click **Options**.

Select **Output Plug-ins** on the left panel to display the available plug-in options. Select **waveOut output v2.0.2a** then click the **Configure** button.
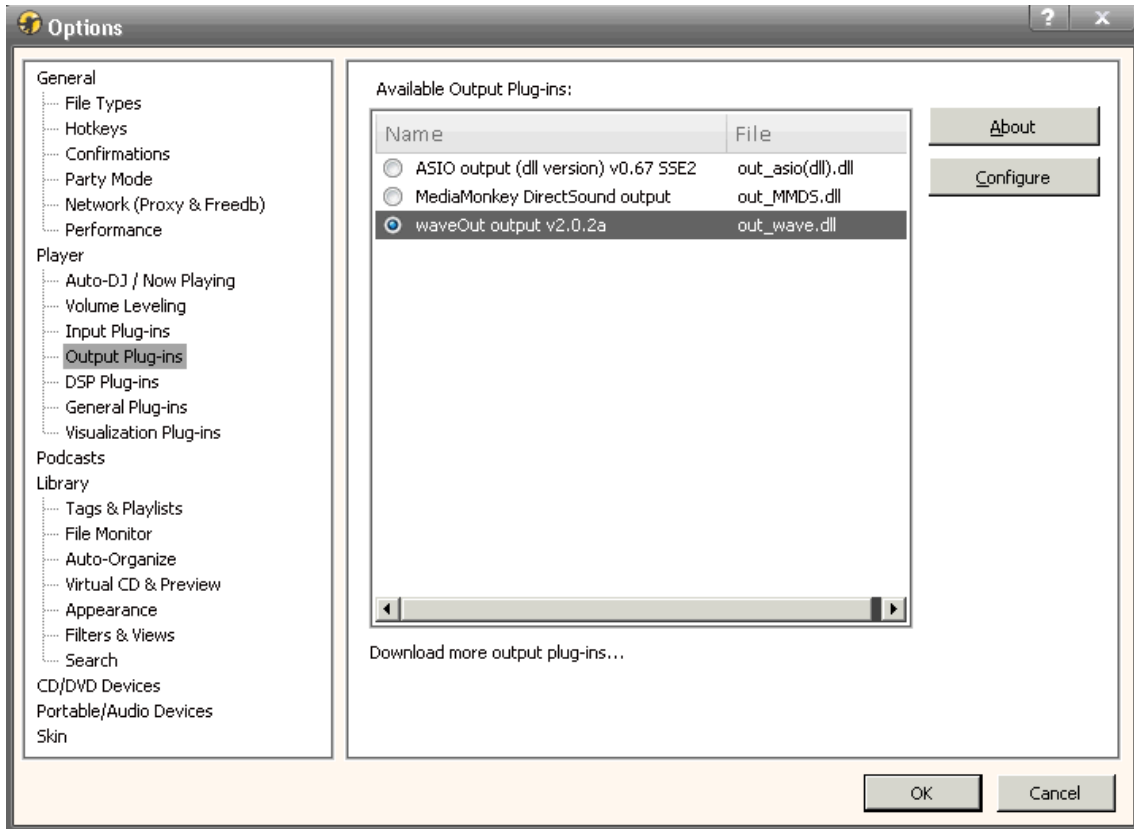
Make sure the correct USB DAC / audio output device is listed to the right of **Device** and that the **Enable** box under **Volume Control** is <u>not</u> checked.
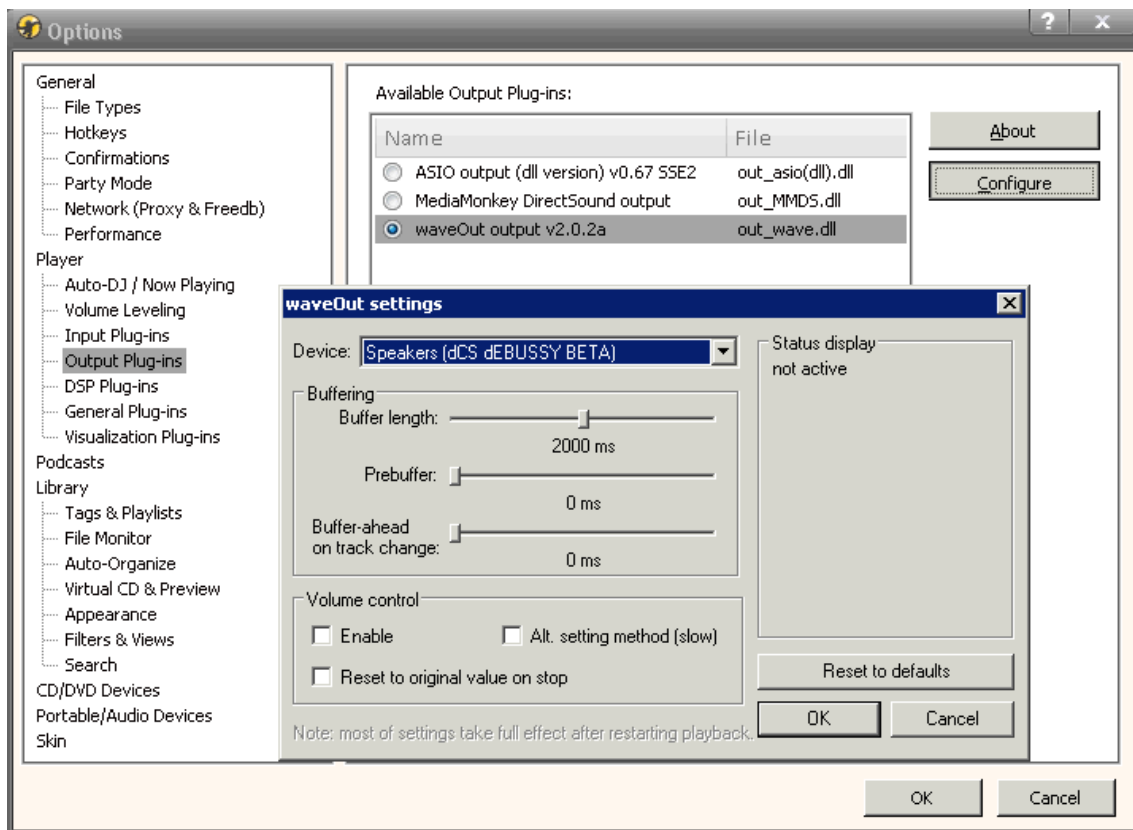
Click **OK,** close MediaMonkey and re-open to conclude the Wave Out configuration.

iTunes Configuration

Download the latest PC version of iTunes from www.apple.com/itunes/download/. iTunes uses a companion program called QuickTime when running under Windows, this is part of the download. Please be sure to update any older installations.

WASAPI

The latest version of QuickTime (7.6) supports WASAPI. WASAPI is not supported by XP, but it is the preferred mode for Windows Vista or Windows 7. The following instructions describe the use of iTunes in WASAPI output mode.

Open **QuickTime**. Open the **Edit** menu, select **Preferences** > **QuickTime Preferences…**

Click on the **Audio** tab. Under **Play Audio Using** select **Windows Audio Session**.
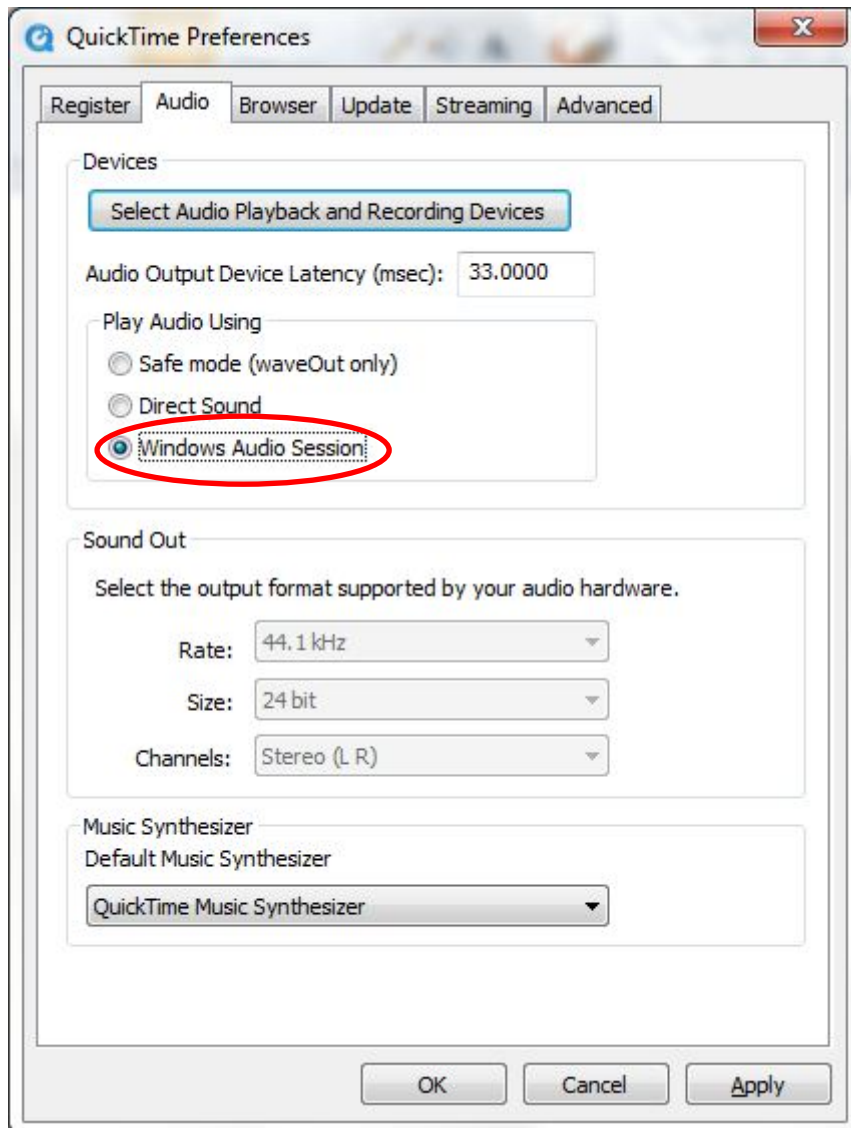
Note that the **Sound Out** settings grey out as WASAPI overrides them.

Click on **OK**, then close **QuickTime**.

Open **iTunes** and play some music.

Direct Sound

Direct Sound mode is described mainly for XP users.

Open **QuickTime**. Open the **Edit** menu, select **Preferences** > **QuickTime Preferences…**
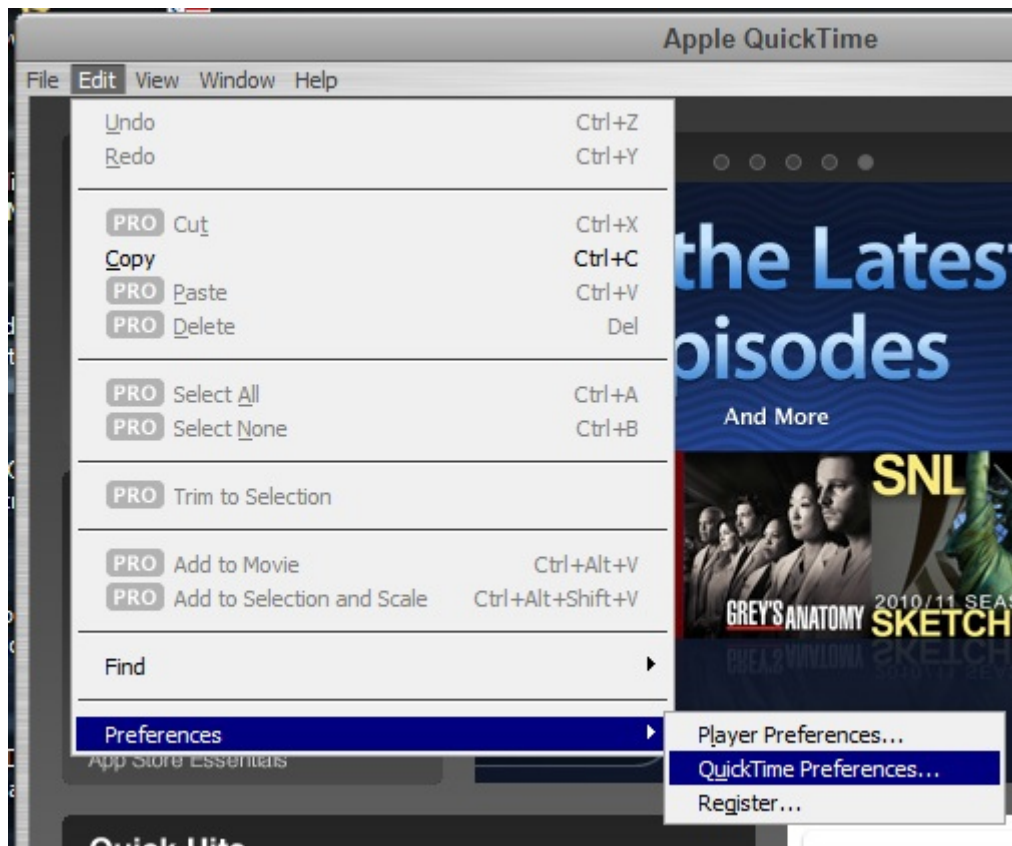
Click on the **Audio** tab. Under **Play Audio Using** select **DirectSound**.

In the **Sound Out** box, set the **Rate** to **44.1kHz**, **Size** to **24 bit** and **Channels** to **Stereo (L R)**.
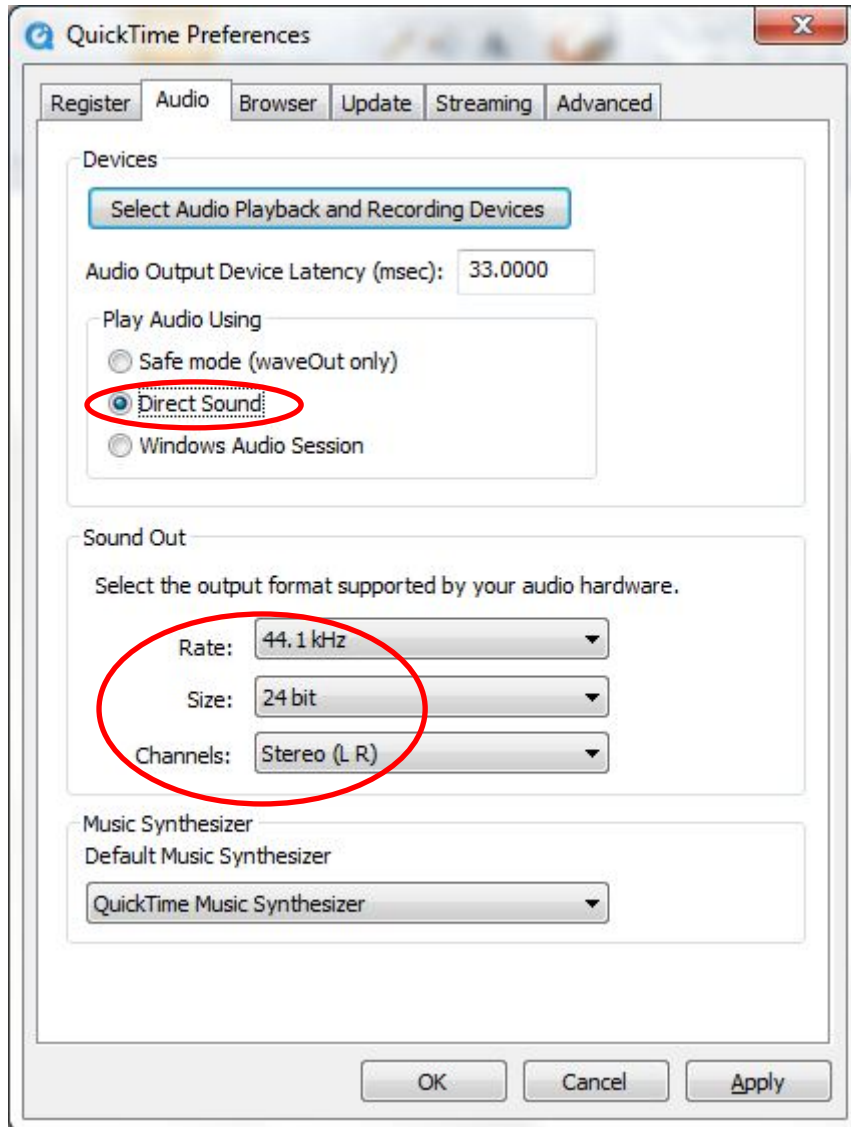
Click on **OK**, then close **QuickTime**.

Open **iTunes** and play some music.

For best results when using files with different sample rates, set **QuickTime**'s **Sound Out Rate** to match the file.

Lynx Studio AES16e Audio Card Configuration

Using an audio card such as the Lynx AES16 or RME HDSP AES-32 may be necessary for the highest sample rates. The following instructions display how to configure a Lynx AES16 in the Windows environment.

Enter the **Lynx Mixer** application.
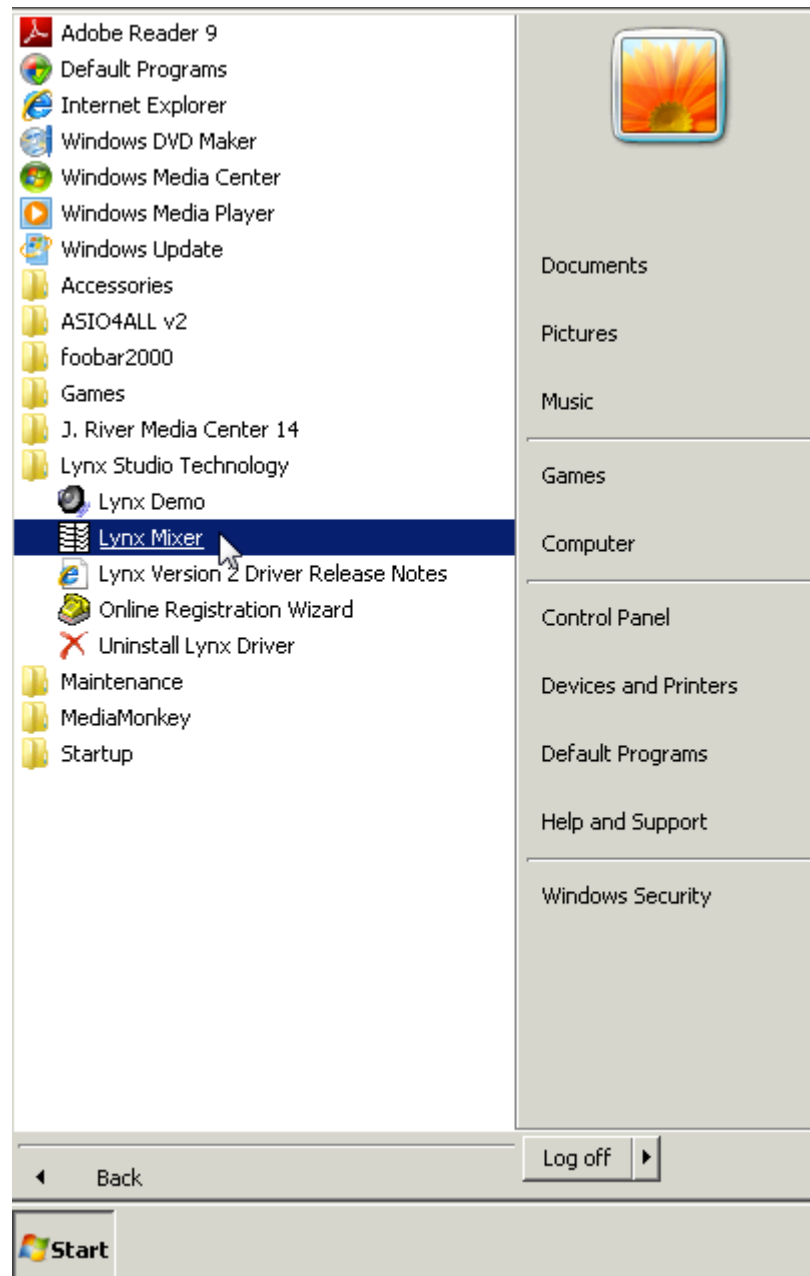
To enable 24/176.4 playback with a DAC that requires Dual wire AES, set **Rate Select** to **176400**.
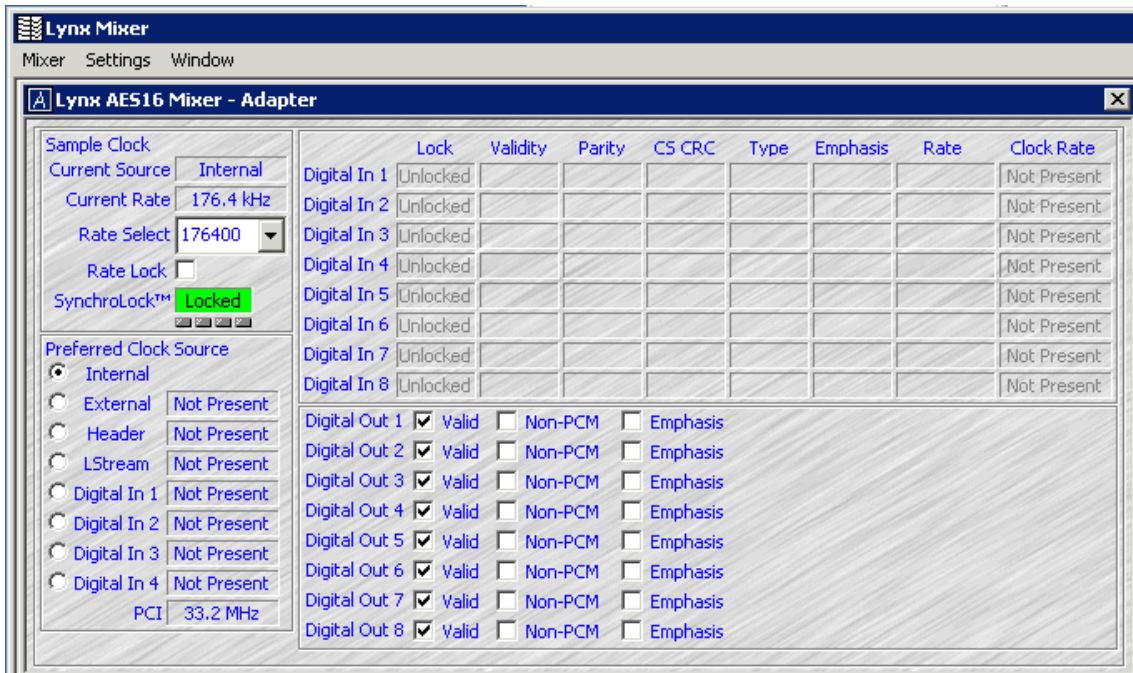
In the **Settings** menu, select the **Dual Wire Out** option then open the **Advanced** menu and make sure **SynchroLock** is <u>NOT</u> enabled.
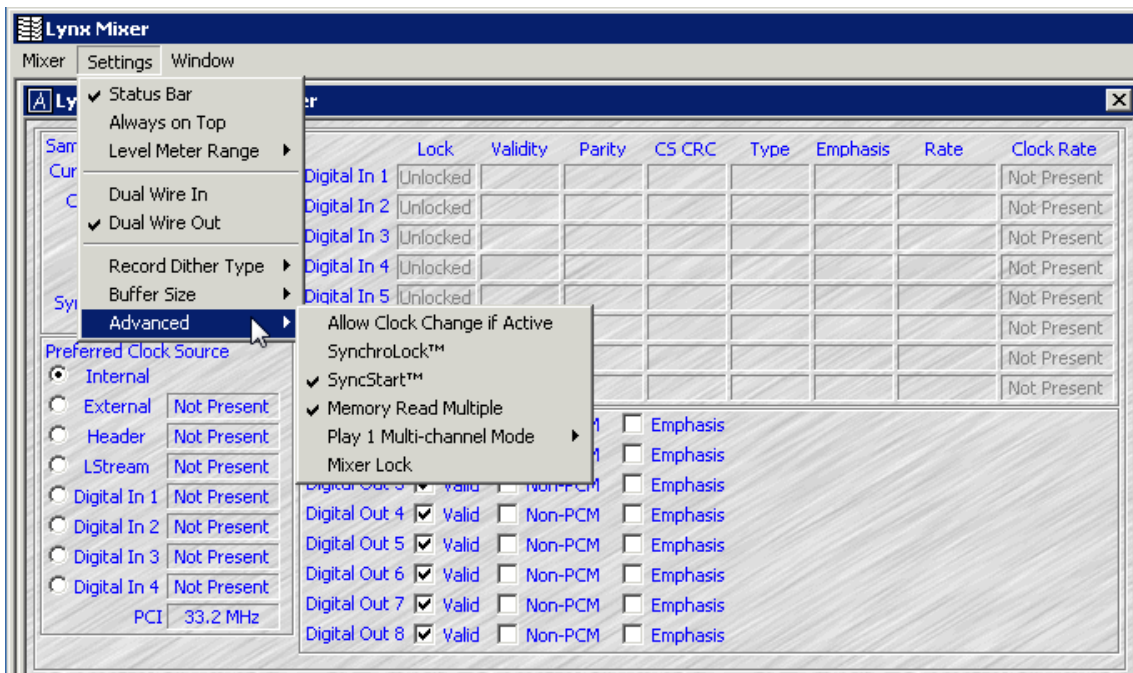
In general, the same application specific configurations apply with a Lynx and a USB DAC. One exception may be large hardware buffers. Within the ASIO **Settings** window there is a checkbox to **Use large hardware buffers** that is sometimes required. It is recommended this option be selected if audio playback is stuttering or there are other playback issues.

VNC Remote Control

VNC (Virtual Network Connection) is an open-source graphical desktop sharing system. It transmits the keyboard and mouse data from the local computer to a remote computer and receives the remote computer's graphical screen updates.

TightVNC may be downloaded free from www.tightvnc.com/download.php. It supports Windows 2000 onwards, Windows Server 2000 onwards and UNIX.

Other versions such as cotVNC support OSX.

Sharing an OSX computer
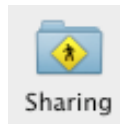
Open **System Preferences**, select ⬛ **Sharing** .

Check the **Remote Management** box. Click the button beside **Remote Management** to turn it **On**. Make a note of the LAN address for this computer as indicated by the green arrow below – **10.20.30.124** in this case.
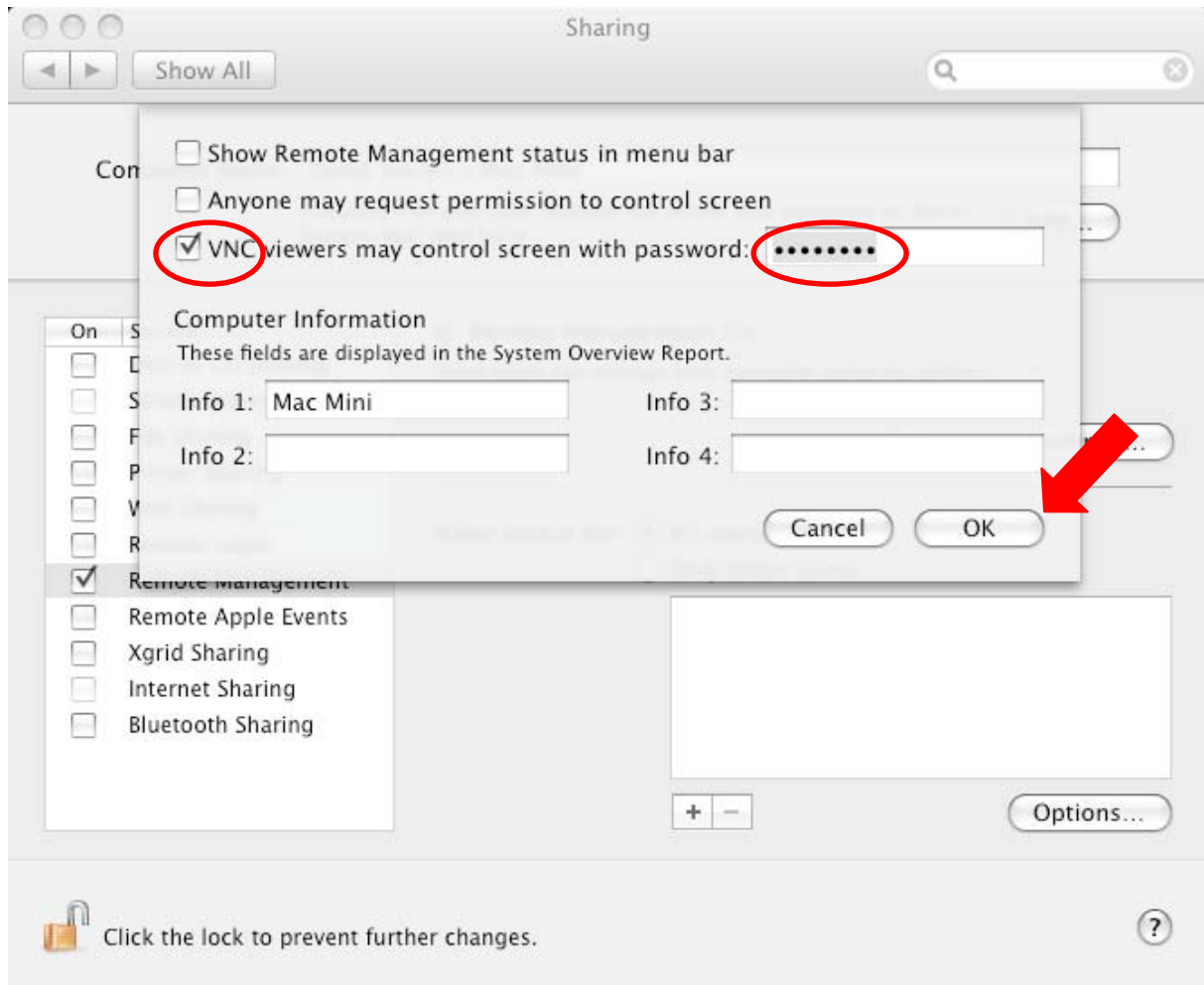
Either **Allow access for: All users** or select **Only these users:** and specify them in the box.

Click the **Computer Settings…** button.
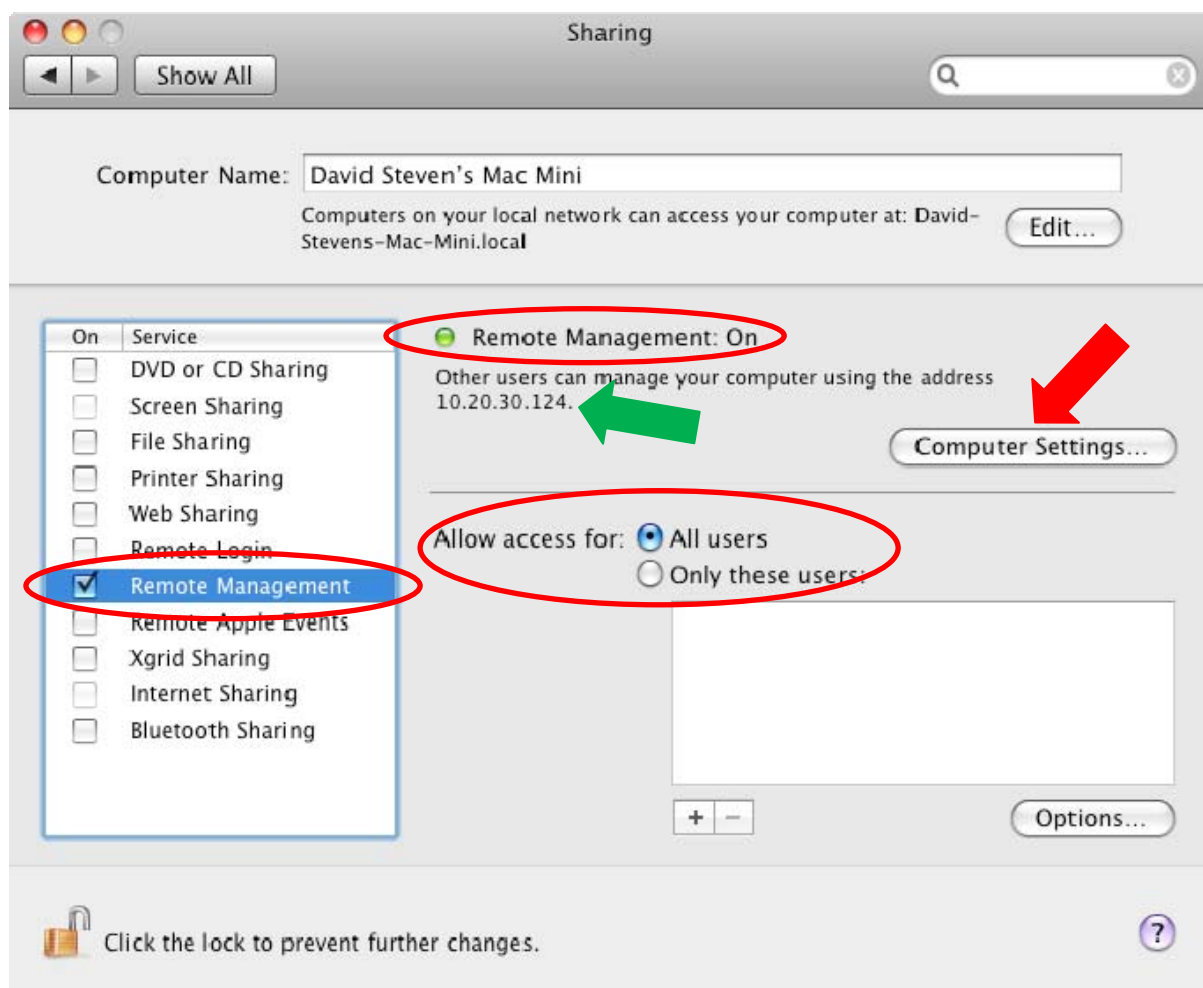
Check the box beside **VNC viewers may control screen with password:** and choose a password.
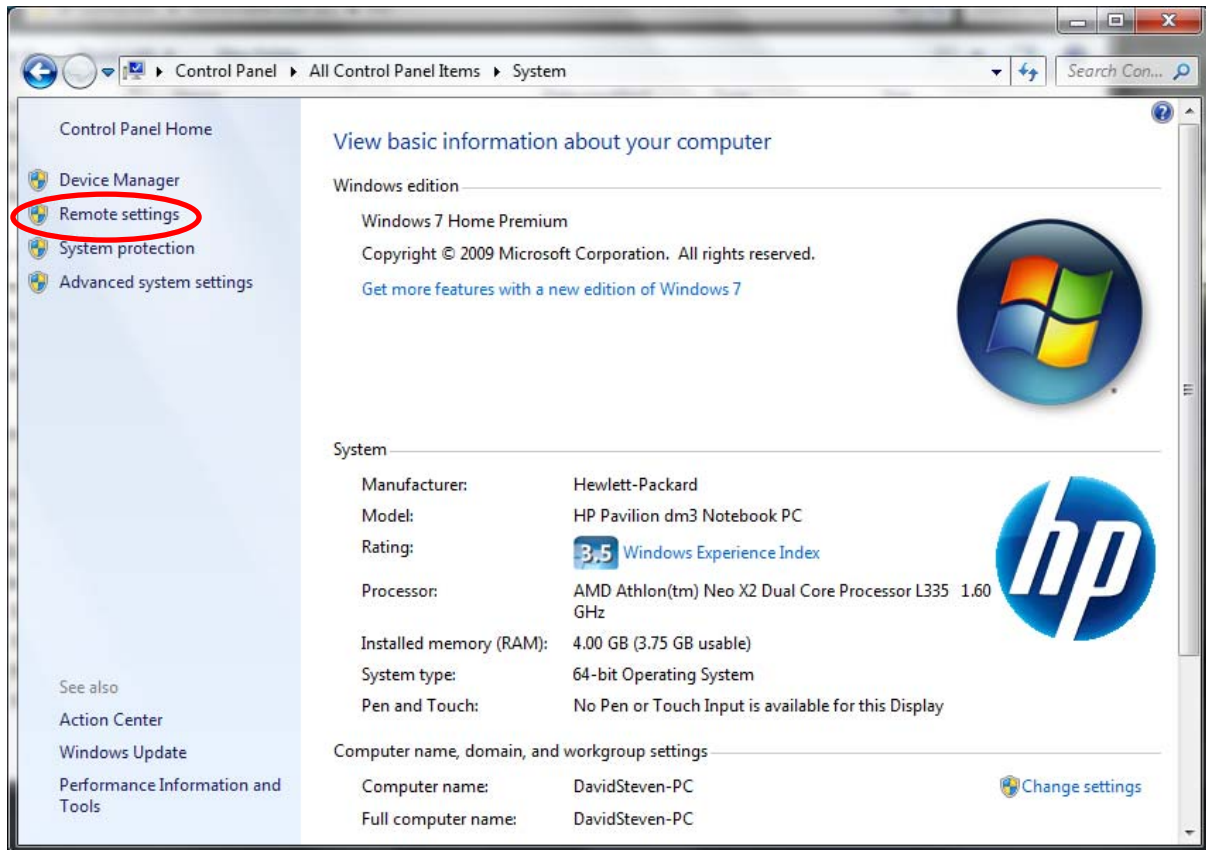
Enter some information in the **Info 1 – 4** boxes to help identify this computer on the network. Click the **OK** button, close the Sharing window.
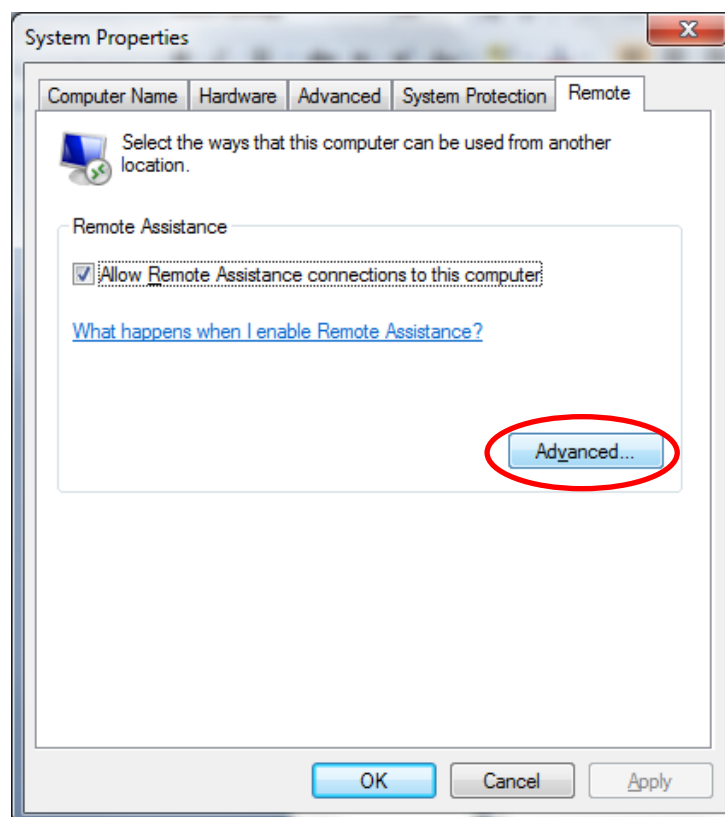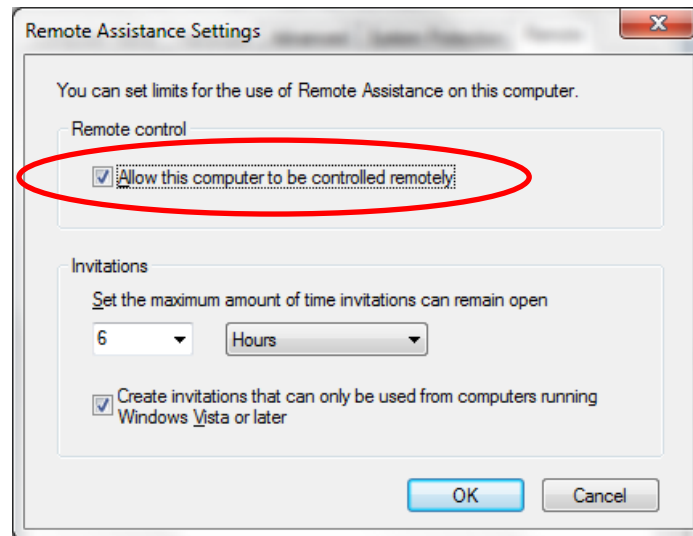
Sharing a Windows 7 Computer

Go to **Start** > **Control Panel** > **System** > **Remote Settings**.



Click the **Advanced** button.

Check **Allow this computer to be controlled remotely** and click the **OK** button.

To find the **IP address** for this computer, click the **Start** button and type **Network Connections** in the search box. Click the active network connection (**Wireless Network Connection** in this case). On the tool bar, click **>>** and select **View status of this connection**.

Click the **Details…** button.
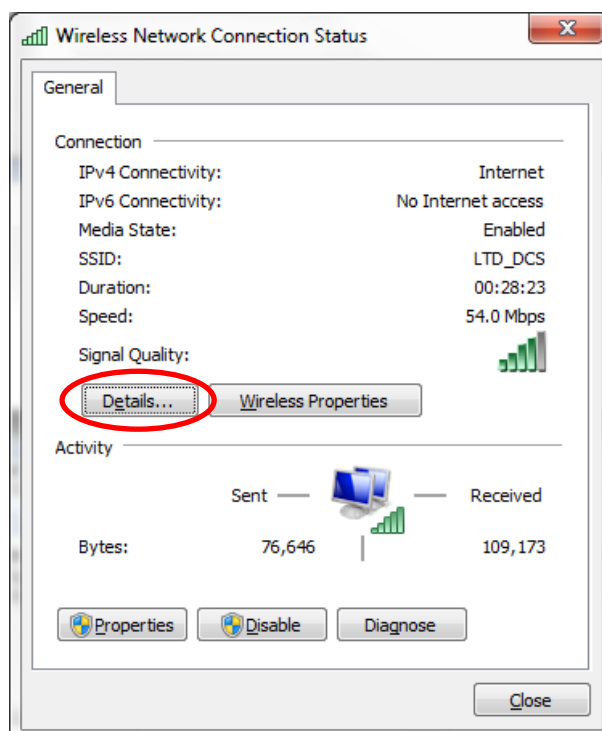
The computer's address is listed beside **IPv4 Address** (10.20.30.148 in this case). Note the address and click the **Close** button.
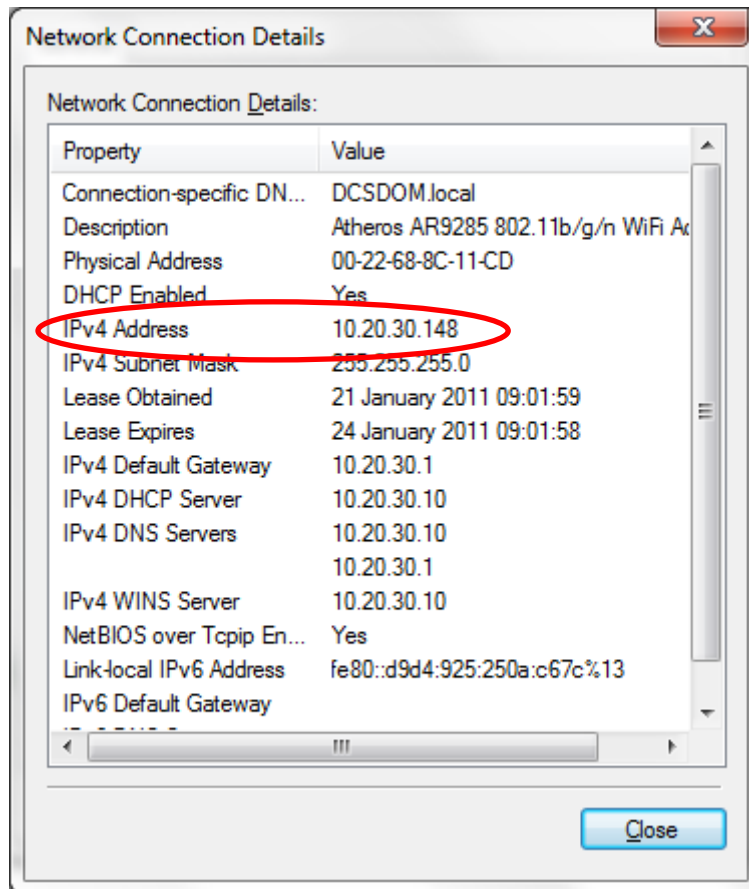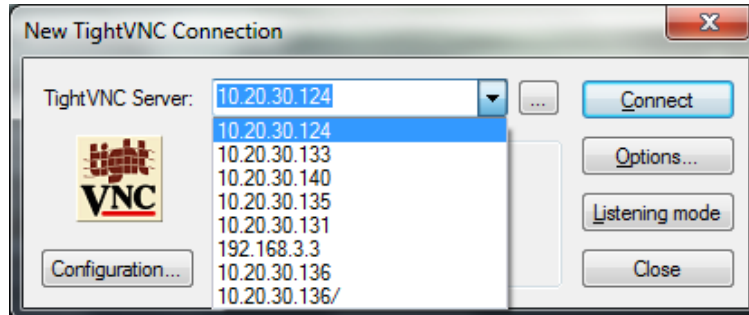
Controlling from a Windows 7 Computer

Download and install **TightVNC** from www.tightvnc.com/download.php.

Make sure both computers are connected to the same network.

Run **TightVNC Viewer**. Select the remote computer's address from the drop-down box beside **TightVNC Server**. (10.20.30.124 in this case.) Click on the **Connect** button.

Type in the **Password** you entered into the OSX computer and click the **OK** button.
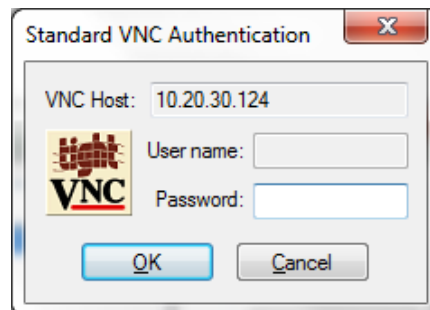
The desktop of the remote computer will appear in a window, you can control it from there.
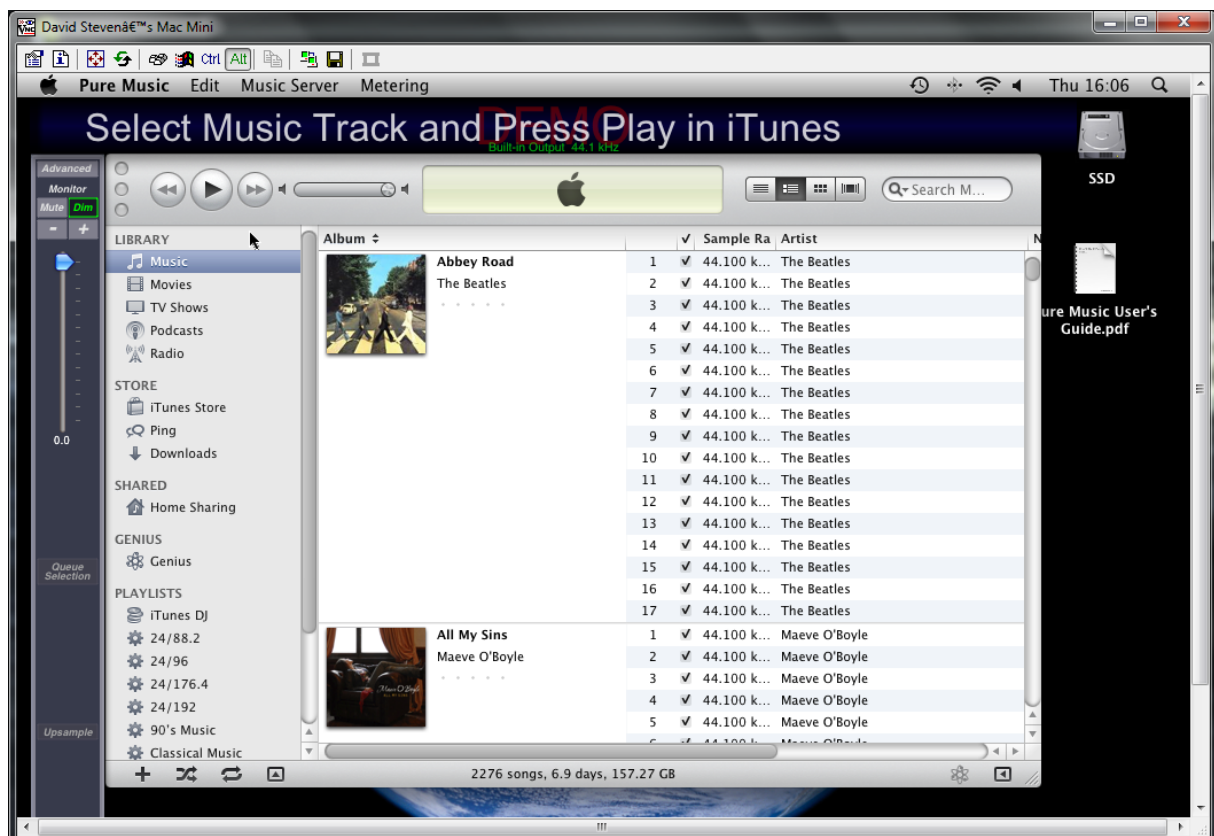
Useful Links

- Detailed Windows 7 & J River Media Center 14 article –
http://www.computeraudiophile.com/content/Windows-7-Audio-J-River-Media-Center-14-Configuration

- Small DIY audiophile music server project –
http://www.computeraudiophile.com/content/Computer-Audiophile-Pocket-Server-CAPS

- Thorough CD Ripping article - **http://www.computeraudiophile.com/content/Computer-Audiophile-CD-Ripping-Strategy-and-Methodology**

- Automated CD Ripping –
http://www.computeraudiophile.com/content/Ripping-CDs-Style-GD3-Review

- Basic articles about computer audio –
http://www.computeraudiophile.com/category/Computer-Audiophile-Academy/Computer-Audiophile-Academy

- For a different viewpoint, take a look at **www.ayre.com/usb.htm**

Music Download Sites

www.linnrecords.com

www.2l.no – high res classical music specialists.

www.iTrax.com

www.highdeftapetransfers.com

www.hdtracks.com

www.deutschegrammophon.com – mainly CD format, but the name speaks for itself.

www.pristineclassical.com – vintage recordings.

EXAMPLE CONFIGURATIONS

Apple Mac Pro desktop based system

Apple Mac Pro
2.8GHz Quad-Core Intel Xeon
8GB 1066MHz DDR3 EEC SDRAM
USB 2.0, FireWire 800

Operating System:
Apple OSX 10.6 Snow Leopard

USB2.0

Solid State Disk
2.5" 100GB SATA2
OCZSSD2-2VTXP100G or similar
Fit inside computer in place of HDD

Playback software:
Apple iTunes
Channel D Pure Music Player
or Sonic Studio Amarra

Puccini U-Clock
Scarlatti Upsampler
Paganini Upsampler

Debussy DAC

5TB Network Attached Storage
NAS drive for main music storage

Optional sound card for 24/192:
RME HDSP AES32
or Lynx Studio AES16

Optional Dual AES
HD26 to AES breakout cable to DAC

Apple MacBook Pro laptop based system

Apple MacBook Pro
2.4GHz Intel Core 2 Duo
8GB 1066MHz DDR3 EEC SDRAM
(4GB SRAM minimum)
A Solid State Drive is recommended for
the OS and applications.
USB 2.0, FireWire 800

Operating System:
Apple OSX 10.6 Snow Leopard

USB2.0

5TB Network Attached Storage
NAS drive for main music storage

Playback software:
Apple iTunes
Channel D Pure Music Player
or Sonic Studio Amarra

Puccini U-Clock
Scarlatti Upsampler
Paganini Upsampler

Debussy DAC

HP Pavilion desktop based system

HP Pavilion Elite PC
Intel Core i7-2600
8GB 1066MHz DDR3 SDRAM
(4GB minimum)
USB 2.0

Operating System:
Windows 7 Ultimate

USB2.0

Solid State Disk
2.5" 100GB SATA2
OCZSSD2-2VTXP100G or similar
Fit inside computer in place of HDD

Playback software:
J River Media Center v15
Foobar2000
or MediaMonkey v3

Puccini U-Clock
Scarlatti Upsampler
Paganini Upsampler

Debussy DAC

5TB Network Attached Storage
NAS drive for main music storage

Optional sound card for 24/192:
RME HDSP AES32
or Lynx Studio AES16

Optional Dual AES
HD26 to AES breakout cable to DAC

HP Envy laptop based system

HP Envy laptop PC
Intel Core i7-720
8GB 1066MHz DDR3 SDRAM
(4GB minimum)
USB 2.0

Operating System:
Windows 7 Ultimate

USB2.0

Solid State Disk
2.5" 100GB SATA2
OCZSSD2-2VTXP100G or similar
Fit inside computer in place of HDD

Playback software:
J River Media Center v15
Foobar2000
or MediaMonkey v3

Puccini U-Clock
Scarlatti Upsampler
Paganini Upsampler

Debussy DAC

5TB Network Attached Storage
NAS drive for main music storage