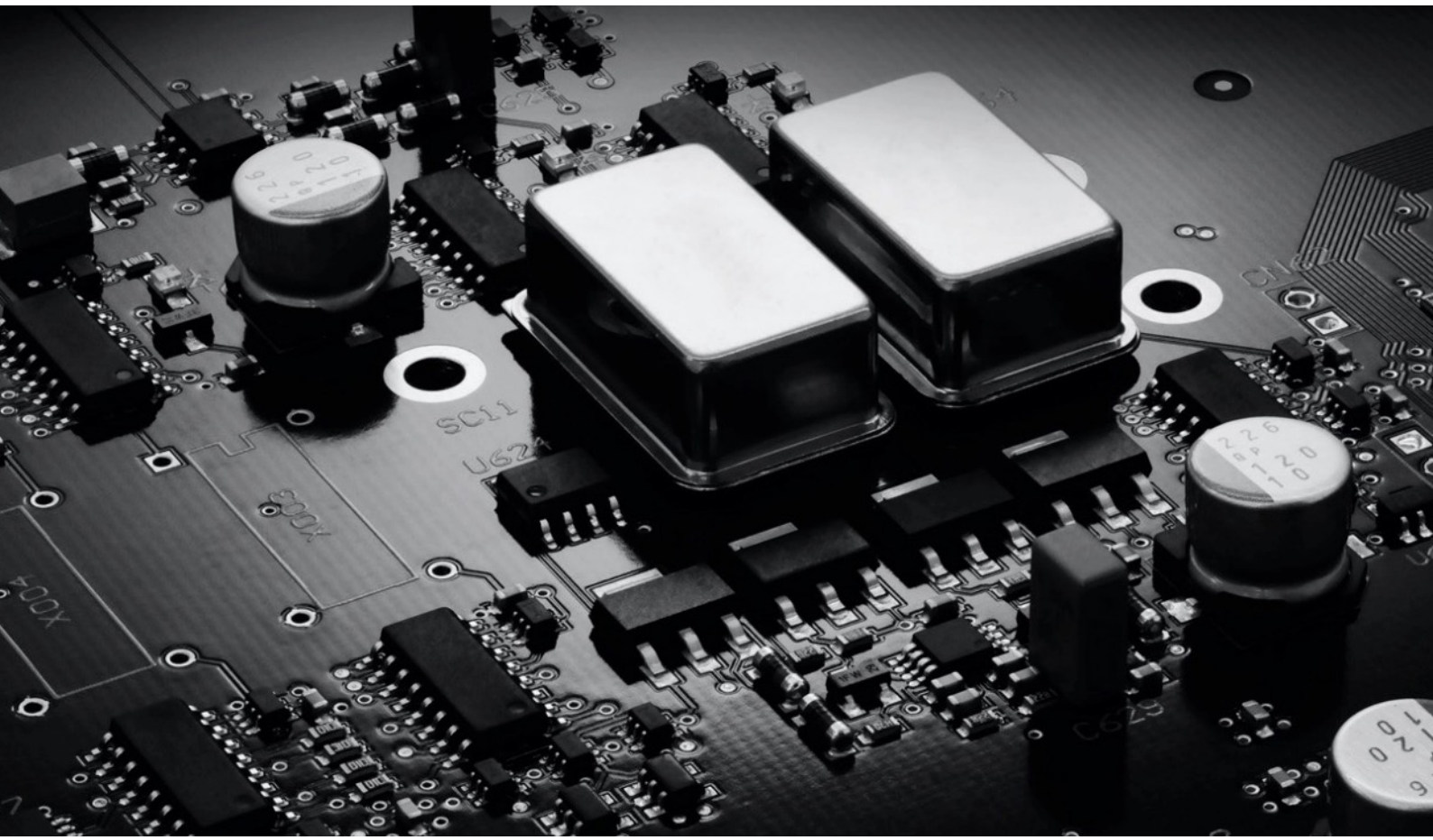
*dCS* Insights & Innovation

dCS
ONLY THE MUSIC

# Clocking in digital audio

## The Critical Role of Clocking in Digital Playback



# Synopsis

In audio electronics, the term clocking refers to the systems used to keep various digital circuits, including power supplies, high-speed signal processing and modulation, in sync with the main audio clock and operating at exactly the same time.

In this paper, we'll examine how we go to extreme lengths to ensure that our clocking system delivers the utmost stability and reliability. All of our internal processing, from power supply synchronisation to the high-speed signal processing and modulation, is synchronous with the main audio clock for consistent behaviour and absolute precision, no matter how you choose to listen.

# Clocking: the basics

Clocking is an integral part of digital audio, and almost all digital audio products have a clock inside. As discussed in the Ring DAC paper, digital audio recordings are made up of a series of samples. An audio product such as a DAC needs to be instructed on when to do something with the samples it receives at its input. This clock provides this instruction.

In the context of digital electronics, the term clocking refers to a signal that synchronizes all the circuits within a system. In order to generate a precise and reliable signal, the clock system must have a source: a reference defining the length of an interval of time. This source usually comes in the form of an oscillator – an electrical circuit that provides a rhythmic voltage cycle.

At dCS, we use quartz crystal oscillators as the basis for our clock systems. Quartz is a piezoelectric material, meaning that when a voltage is applied to it, it physically deforms and flexes back and forth. The crystal can be designed to resonate mechanically at a particular frequency (for example, at 44,100 Hz). With a correctly designed electrical circuit, this resonance can be converted into an oscillating voltage.

The frequency of the crystal's resonance provides the reference for an interval in time—such as 1/44,100th of a second. A digital system can use these precise intervals to reference the accurate spacing of samples. This avoids any unwanted movement of the samples in time. When the samples are placed incorrectly in the time domain the process of converting digital to analogue, the errors cause measurable and audible distortion of the audio signal.

The design of clocks in digital audio (both internal clock systems and external master clocks) is a topic worthy of serious consideration when purchasing any high-end audio system. Arguably, clocking can have as much of an impact on sound quality as a high-quality DAC. It's therefore vital to consider the design and implementation of the clocking systemically.

As the clock defines the timing of a DAC's operations, it is responsible for ensuring the samples are converted at the correct time, which is crucial to ensuring the audio we hear during playback is natural and accurate.

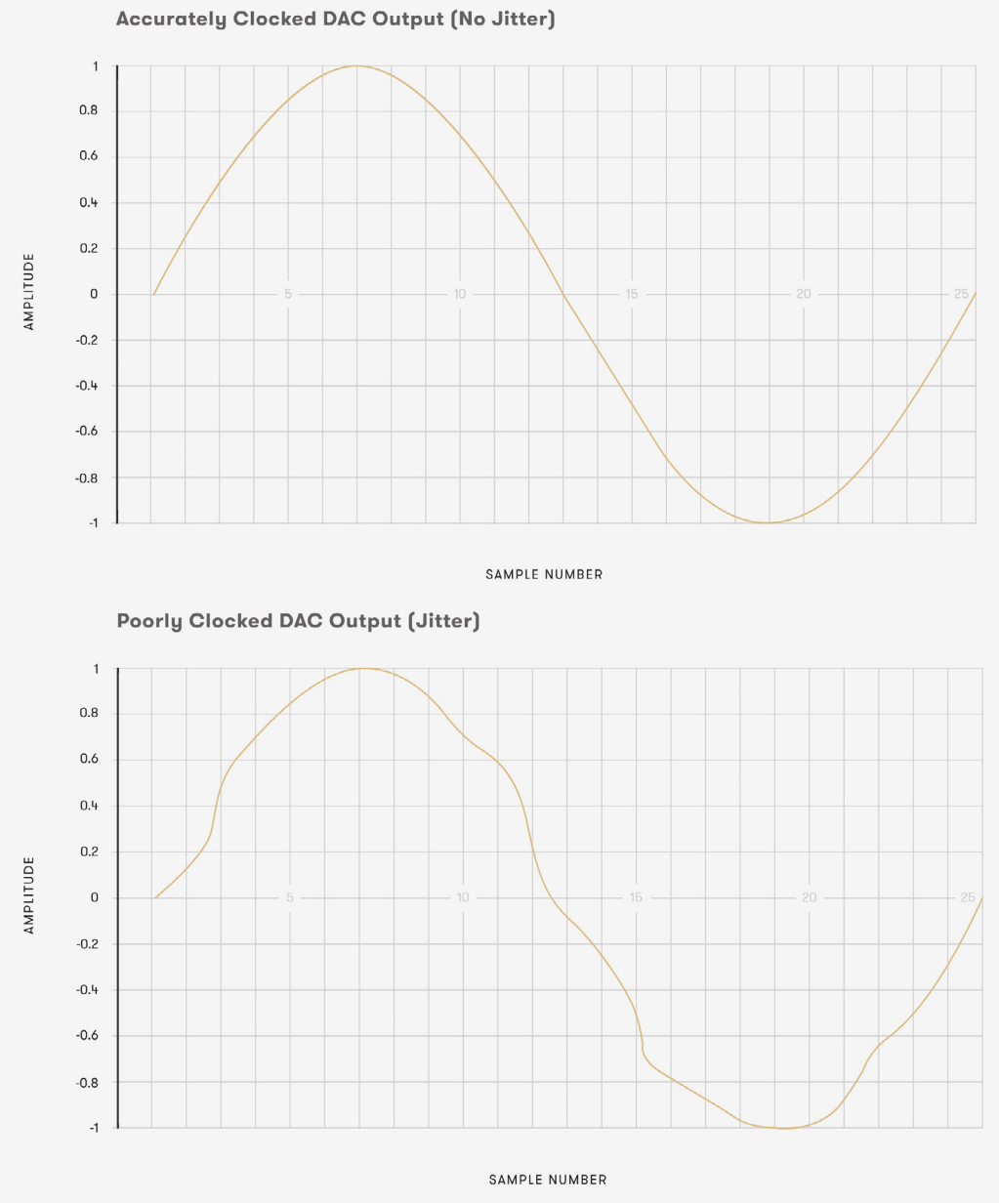
# Jitter

If a clock system fails to produce time-reference signal correctly during the D/A conversion process, or the signal is unable to reach a DAC, the result is type of signal error called jitter, which is highly undesirable in the accurate reproduction of music.

Jitter is described as any irregularity in the timing of the clock used by a DAC. It is produced in a variety of ways. It can be the result of inadequate analogue design, electromagnetic interference, a poor-quality digital audio cable, or several other causes. We will discuss these causes in this paper.

> "The human ear and brain are extremely sensitive to irregularities in the time-domain."

The actual audible effect of jitter depends upon its nature, but it can have a substantial impact on sound. If jitter is periodic, sidebands will appear either side of the signal frequency. This sound is experienced as harshness, as artificial components and distortion are being added to the audio. If the jitter is noisy in nature, this results in a "smearing" of signal energy. This, in turn, increases the noise floor of a system, which has the effect of masking fine detail in the music.




The above graphs show effects of poor clocking on even simple signals. In both examples, a sine wave has been reconstructed by a DAC using 25 samples. Each of these samples are the same frequency and amplitude; the only factor that has changed is the timing of the samples during the conversation process. The result is a visible (and audible) degradation of the signal. Were this signal to be played back through a transducer, the signal in the lower graph would exhibit audible distortion caused by the jitter.

While the example above is rather exaggerated, it illustrates that the right sample at the wrong time is the wrong sample. If musical accuracy is desired, accurate clocking within a digital audio playback system is vital.

The human ear and brain are extremely sensitive to irregularities in the time-domain. If a DAC experiences jitter and fails to convert signals into analogue voltages at the correct time, then the sense of space in a performance can be heavily skewed or even lost. The harmonic structure of musical instruments is degraded. The sense of natural ease and effortlessness present in live music is absent. It's for this reason that dCS engineers take great care to minimise jitter in all aspects of our design.

If jitter is introduced at the recording stage, it will remain in the signal forever. There are steps that can be taken to prevent further degradation of the signal (such as re-clocking or even buffering the signal into RAM and out again) but it isn't possible to correct or remove jitter introduced during the recording process.

At the playback stage, it is extremely desirable to reduce jitter wherever possible. Provided a recording is of good quality, a DACs ability to reproduce audio samples precisely at the correct interval in time substantially increases its ability to create an accurate representation of the original sonic event. A signal coming into a DAC from an external source, such as a CD transport, often exhibits irregularly spaced samples on arrival. This is true of other digital sources as well. Provided the DAC itself possesses the ability to convert those samples at regular intervals, the sound quality will remain unaffected.

# Jitter (intrinsic)

There are two major types of jitter—intrinsic and interface. Intrinsic jitter refers to jitter which is produced inside of a product like a DAC, through effects like phase noise on the oscillator. Interface jitter refers to jitter which is picked up by the interface(s) used to transfer the audio and clock signals. This could come either as interference picked up by the cable itself, or through the cable essentially acting as a filter for certain frequencies, impacting the integrity of the square wave (the output of the clock circuitry) passing through it.

> "...we eschew any approach that compromises the quality of a DAC's clocking to allow for a sub-optimal source to be connected and work properly"

There are several varieties of quartz crystal oscillators, but Voltage Controlled Crystal Oscillators (VCXOs) and Oven Controlled Crystal Oscillators (OCXOs) are two of the most common in audio.

Voltage Controlled Oscillators, or VCOs, are also used in audio products, but these operate on a purely electronic basis, and do not use an electromechanical material such as quartz to generate signals.

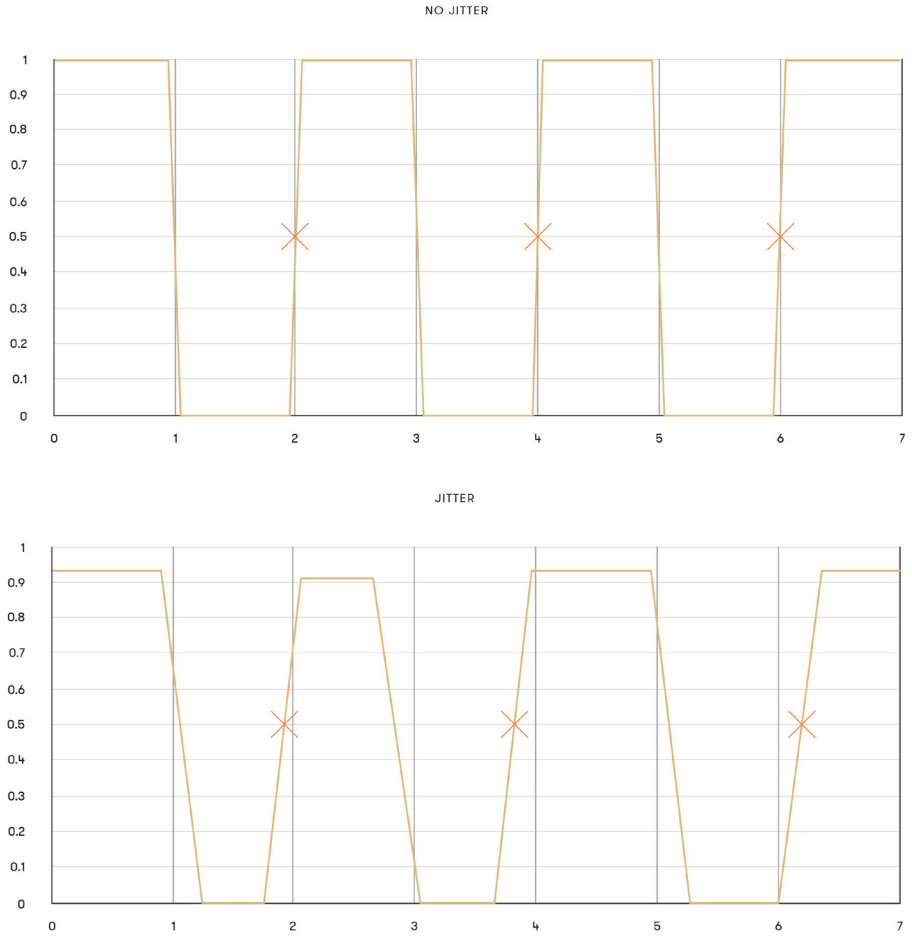
Quartz oscillators tend to have better phase noise performance than VCOs, meaning the oscillator itself is less prone to jitter. What this means in terms of overall clock design is that in a DAC with a quartz oscillator, the Phase Locked Loop, or PLL (the circuit which matches the frequency of the DAC's clock to the clock of the incoming audio signal) can be biased towards rejecting interface jitter by means of a narrower PLL bandwidth.

This is possible as the quartz oscillator itself is less prone to phase noise that causes jitter. As such, if jitter is present on the interface, for example a jittery AES signal, this jitter will not be passed on to the DAC, as it will have come and gone before the PLL reacts to it. The DAC instead relies more on the oscillator for timing accuracy between individual samples which, in the case of a dCS product and its quartz crystal-based clock, is a very high level of accuracy.

The alternative to this would be to use a VCO as the oscillator. However, given the poorer phase noise performance of a VCO compared to a quartz oscillator, the PLL within the product may need to be biased towards rejecting intrinsic jitter, as the oscillator itself would likely be more prone to phase noise. This is achieved through using a wider bandwidth PLL. Any interference or cable filtering effects will have a more direct impact on the sound.

If this is the case, you might be wondering why any product would use a VCO as a clock source. One benefit of using a VCO over a quartz oscillator is the possibility for the clock to have a greater "pull range", meaning it can lock to a wider range of signals (for example, signals running consistently too fast or too slowly).

Since dCS is focused on designing the most accurate digital audio playback systems, we eschew an approach that compromises the quality of a DAC's clocking to allow for a sub-optimal source to be connected and work properly. At dCS, we opt for using a quartz-based oscillator with a high level of accuracy and stability, allowing for a pull range of +/- 300 parts per million (PPM), as per AES specification.




*The graphs on the previous page show the effect of jitter on the square wave output by a clock circuit. As previously discussed, jitter impacts the transition times of the wave and the peak voltages the wave can reach. This has the effect of changing the point in the time domain at which the system would perceive, for example, a 0 changing to a 1. Clock systems watch the "rising edge" of the clock signal where the voltage increases, so the point on the rising edge where the amplitude goes above 0.5 has been marked on the graphs. The timing of this is regular in the first graph – the transition points on the rising edges fall on 2, 4 and 6 on the X axis respectively. When jitter is introduced, the transition points are shifted forwards or backwards depending on the nature of the jitter. It is not regular; it is random.*

There are several factors that can cause phase noise (and, consequentially, jitter) on an oscillator, and these should all be considered when designing a clock system.

# Physical Vibrations

As the basis of a quartz clock is its piezoelectric property (the physical movement of the crystal when a voltage is applied), any external physical vibrations can cause inaccuracy in the clock. The extraneous movement does not need to be vigorous to cause inaccuracy. It could be as subtle as, for example, the vibrations of a CD mechanism inside the product. Any measures which can be taken to isolate the clock circuitry from the external physical vibrations should be taken, as this creates a higher level of clock performance.

# Power Supply

The ability of a quartz crystal (or any piezoelectric material) to maintain a consistent oscillation frequency relies a stable, interference-free DC signal of the correct specification. In the case of both VCXOs and OCXOs, this means clean DC for the power supply. In the case of a VCXO, it is even more crucial that the control voltage is extremely stable. In this type of oscillator, the control voltage is used to make fine adjustments to the crystal's frequency. If there is any variation in the power rails fed to the crystal, the resonance frequency will change. In products with a quartz-based clock, designers should always endeavour to provide as clean a power supply fed to the crystal(s) as possible, at the correct voltage and frequency for the specification of the region.

# Crosstalk

Electronic circuits can generate electromagnetic leakage. This is often seen when running high-rate signals, such as digital audio, through the copper tracks found on printed circuit boards (PCBs). The copper tracks essentially act as antennae, with the digital audio signals being radiated from the board. This interference can affect associated clock circuits if they are in proximity, negatively impacting the performance of the clock.

The correct way to eliminate this problem is to design the product's PCBs in such a way that minimises crosstalk. Secondary to this is to ensure that any sensitive parts are separated from those which may cause interference. An even more effective method is to completely remove many potential sources of electro-magnetic (EM) interference from the product where possible. Using Master Clock—a standalone clock with its own dedicated circuitry and power supplies—is the most effective approach of all.

# Clock Frequency

The ideal way to design the clock inside an audio product is to have two oscillators: one running at direct multiple of 44.1kHz, and the other running at a direct multiple of 48kHz. The reason for this is that almost any sample rates used in digital audio are multiples of these “base rates” (including DSD, which runs at very high multiples of 44.1kHz). If the clock does not use direct multiples of the sample rate it is to be clocking, the maths becomes more complex and the electronics that need to be used to generate the correct frequency are more prone to jitter.

Trying to clock a 44.1kHz signal with a 10MHz clock, for example, would require somehow synthesising 44.1kHz from 10Mhz, which is mathematically inefficient. As such, this type of clock will need to use methods such as asynchronous rate conversion to multiply the rate down correctly. These methods invariably result in a “dirtier” frequency spectrum of the clock signal, meaning that the system will be more prone to jitter.

dCS products use two oscillators, running at 29 of the base audio rates (44.1kHz and 48kHz), so 22.5792MHz and 24.576MHz. The easy division down to any required rate results in a cleaner clock spectrum and, as a result, significantly less jitter.

# Clock Temperature

While clock temperature is not a source of phase noise, it can affect the performance of an oscillator. The resonant frequency of a quartz crystal is inversely proportional to its size and, by extension, its temperature. As the temperature of the quartz increases, it physically expands. As the temperature decreases, it contracts. This causes changes in the resonant frequency of the quartz. Temperature variations in digital systems should therefore be avoided, or the effects mitigated wherever possible.

There are several methods for counteracting temperature variations inside a crystal oscillator. One approach is to use an OCXO. An OCXO aims to remove the temperature variation of the crystal by using a Curie heating element to keep it at a stable temperature. The Curie device is a resistive heater whose resistance increases sharply when it reaches a certain temperature, effectively limiting the heating power. The temperature will overshoot and then stabilise around the required temperature. When the temperature of the product is not stable (such as when it is powered on from cold), there will be some fluctuation in the temperature due to thermal delays and, consequently, the frequency as the system “hunts” for target temperature. Once the crystal’s temperature has stabilised, however, the clock will output a stable frequency.

Another approach is to use a microcontroller-enhanced VCXO, which dCS employs in a number of our products. This approach does not use any heating elements to account for temperature variation. Instead, we utilise the large amounts of processing power available in our FPGA-based design processing platforms to continuously adjust to the control voltage fed to the VCXO to compensate for temperature changes.

In the case of a dCS Master Clock, such as the Vivaldi Clock, these adjustments are based on intensive measurements taken during production. During the production process, we place the clock (and the circuit board the clock is fixed to) into an Environmental Chamber. This chamber measures the clock frequency against the current controlled environmental temperature and records it onto the FPGA inside the product. The temperature is then changed, the clock measured, and the performance again logged. This process is repeated over 18 hours. This enables us to plot exactly how the VCXO in the Master Clock behaves at any given temperature—data which is stored on the product's FPGA.

The data is utilised in the product by adjusting the control voltage which is fed to the VCXO. A higher or lower voltage will create a higher or lower resonant frequency. This, combined with the product's "knowledge" of its performance against temperature, ensures the clock's output frequency is always extremely stable. At any given normal operating temperature, the clock's output frequency will be consistent.

Rossini and Vivaldi Clocks utilise this process to continuously measure clock temperature, and instantaneously adjust the control voltage if the clock temperature has varied. The result is that a new Vivaldi Clock, for example, can achieve an accuracy of above +/- 1 PPM when shipped. Once the clock has stabilised in its environment, the accuracy actually improves over time, typically by as much as an order of magnitude: +/- 0.1 PPM.

# Jitter (interface)

If a product is locking to the clock signal of an external source, such as a CD transport connected to a DAC, interference picked up by digital audio cables between the products can smear the transition times of the clock data within the signal—essentially, changing the point in time where a 0 changes to a 1, or vice versa.

"It's important to ensure that a cable's bandwidth is adequate for digital signals"

Balanced connectors help reduce interference introduced in the cables. This is why the AES/EBU format uses 110Ω twisted pair, shielded cable. This geometry effectively shields the conductors in the cable from most electromagnetic interference (EMI) and directs any that it picks up to ground, eliminating it from the signal. Any EMI that gets through to the conductors gets phase-cancelled, because each conductor is exactly 180 degrees out of phase with the other.

It's important to ensure that a cable's bandwidth is adequate for digital signals. The square waves carrying the signal have a very fast rise time between the low and high states (the 0s and 1s). A fast rise time translates to a very high frequency—in the megahertz range. For this reason, it's advisable to use good quality cables that are specifically designed to carry digital audio data—110Ω cables for AES transmission and 75Ω cable for S/PDIF transmission.

When a digital signal is passed through a cable, it will, to a degree, act as a filter. A poorly designed cable, one unfit for use with the interface it is designed for could potentially filter out high frequencies from the signal before it reaches the DAC.

This causes an interaction between any two consecutive data bits within the signal, called intersymbol interference. Depending on the relationship between the first and second of any two bits, the transition between the two can be temporally smeared. The ideal clean vertical line of the square wave becomes more sloped, meaning the exact moment a 0 changes to a 1 or vice versa is blurred. Jitter is often introduced purely from the interactions within the data itself.

If the timing data in the audio signal is being used to lock the DAC's clock to the source's clock, this intersymbol interference will have a negative impact on sound quality, as it can introduce jitter to the DAC's clock. However, if the audio system employs a Master Clock, the timing information embedded in the AES3 signal is no longer being used. The effects of intersymbol interference are negated. While the same filtering effect in the cable and interactions within the data occur, the intersymbol interference does not cause jitter. This is because the Word Clock signal being sent from a Master Clock is regular and does not change like an AES signal does.

It is worth noting that as the PLL used in a dCS product is slow acting, and the clock recovery circuits used are very capable, the effects of intersymbol interference are minimised in cases where the DAC needs to lock to the clock information embedded in the audio signal (such as instances where a Master Clock is not available).

# Clock Synchronisation

There is a problem posed when multiple digital audio devices, each with their own internal clocks, need to work together. One example is feeding a CD transport into a DAC. The DAC has a buffer – a section of temporary memory which stores the audio samples it receives from the CD transport. The transport's clock dictates when a sample is sent out to the DAC, and the DAC's clock dictates when the sample is used and converted to an analogue voltage.

> "In an ideal world, the clocks in the DAC and transport would be running at the exact same rate with no time variations."

In an ideal world, the clocks in the DAC and transport would be running at the exact same rate with no time variations. In reality, however, there are always variations in the clocks (potentially caused by the intrinsic jitter factors discussed earlier). This poses a problem somewhat different to jitter.

If the clocks are running at different rates, on average, over a long period of time, and are left to their own devices with no method of synchronising the two, there will come a point where either the buffer in the DAC has used all the available samples from the transport. This happens when the transport is sending the samples too slowly and the DAC is using them too quickly, or the buffer overflows because the transport is sending samples too quickly and the DAC is using them too slowly. In either event, the this causes temporary audio dropouts, as the DAC must drop everything and relock to the audio signal to get audio samples flowing properly again.

There are two main ways to address this issue. Firstly, there are pieces of timing information embedded within the digital audio signal that the transport gives out in S/PDIF or AES format. The DAC can look at this timing information and adjust the speed of its own clock to match. This means the clocks of the source device and DAC will now be running at the same rate, so dropouts will no longer occur.

The second method that can be employed is to lock both the source and DAC to a Master Clock. A Master Clock is a unit which sits external to all other units in a system and provides a clock signal, referred to as Word Clock, to the rest of the system. The internal clocks of all other units within the system can then be locked to this signal, meaning that they are running at the same rate as the Master Clock. This means that at no point should the DAC suffer from dropouts or re-locks due to the buffer under or overflowing, as on average, the samples are being sent from the source device at the same rate as they are being consumed by the DAC.

The common factor between these two methods is that they both require a method of synchronising an incoming signal with the product's internal clock, by way of a PLL. There are several DACs in the high-end market that do not have the ability to match their clock domain to that of an incoming source, as the oscillator(s) run at a fixed frequency. This means that the unit will drop or repeat samples every now and again, which is very undesirable behaviour. These DACs also suffer from variable latency, so cannot be used for video because of the resulting lip-sync drifts.

As an aside, it is worth noting that the use of a Master Clock in a dCS system does not replace the internal clock inside of the DAC. It simply acts as a stable reference for the DAC to lock itself to, and allows the DAC and source to be properly synchronised without issues such as intersymbol interference causing jitter within the audio data. The DAC's internal clock still dictates when samples are converted, it simply adjusts its frequency over time to match that of the Master Clock. This means the DAC still benefits from having a high-quality clock close to the DAC circuitry. The clock directly controlling the audio is still part of a tightly controlled environment, while simultaneously maintaining sync with the rest of the system.

# Phase Locked Loops (PLL)

A Phase Locked Loop, or PLL, is a circuit that works to match the frequency of an incoming signal with that of an outgoing signal. They are often used to synchronise a DAC's internal clock to that of an incoming signal, such as SPDIF from a CD transport. A "phase detector" in the PLL attempts to match the phase of the incoming SPDIF signal with that of the DAC's internal clock. Its aim is to reduce phase error as much as possible, ensuring that, over time, the two clocks run at on average the same rate, and the DAC's buffer never under- or overflows.

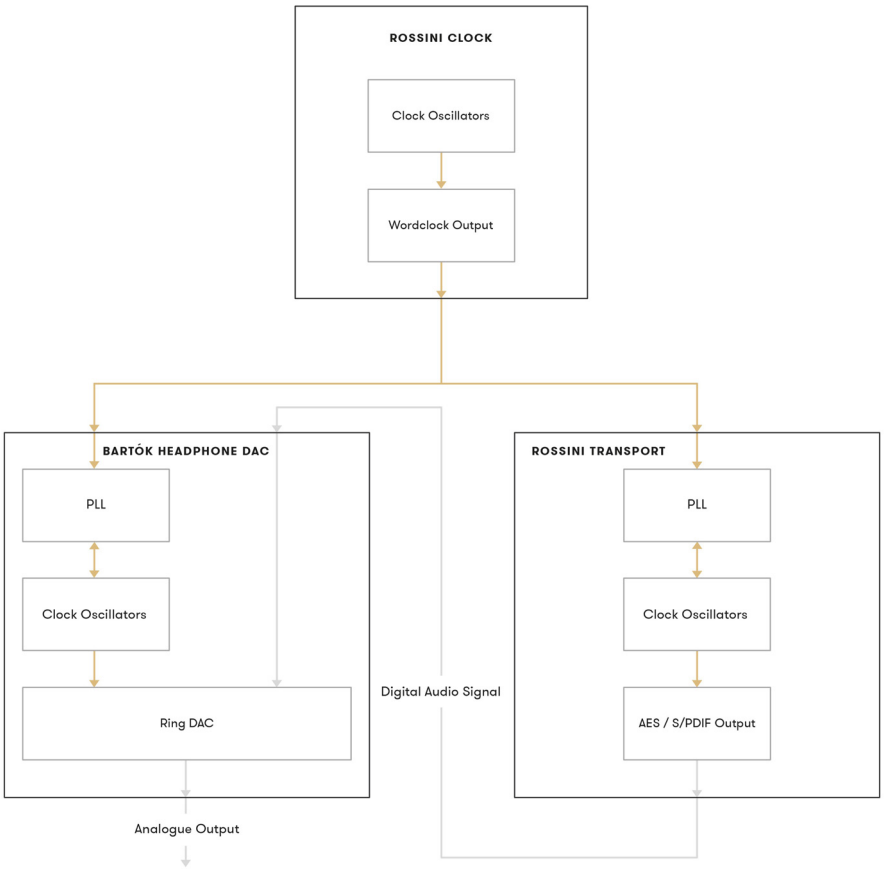
> "The programmability of our platforms gives us a tremendous amount of flexibility in terms of how we facilitate nearly every function of the product."

PLLs are commonly used in audio products with "off-the-shelf" SPDIF receiver chips. This type of chip is typically utilised on the SPDIF input of a product, combining an SPDIF to I2S block together with a PLL. Using a third-party solution such as this often causes problems. With such a chip, it can be very difficult to separate out the functions of signal conversion and clock domain matching. This becomes problematic when attempting to use a Word Clock signal as the clock master for the DAC. AES clock extraction is a good example. This digital function is quite difficult to do well. The structure of illegal codes within the signal means it is easy to induce jitter from the channel block marker that occurs every 192 samples (the structure of SPDIF/AES is beyond the scope of this paper but in essence, the signal deliberately breaks the "rules" by having periods of 3 0s or 1s in a row for various reasons, including to lock the PLL to).

At dCS, we've taken a different approach. dCS DACs still use a PLL, but it is a hybrid design, developed entirely in-house. Part of the PLL is digital, by way of DSP inside the product's FPGA, and part of it is analogue. This lends an enormous amount of flexibility, and a much higher level of performance. Additionally, it is completely independent from the input source. We are also able to carry out functions like dramatically altering the bandwidth of the PLL. This allows the DAC to lock very quickly to a source, thanks to a wide bandwidth on the PLL. The bandwidth can then be tightened over time to reduce jitter.

This approach ensures that the clock and data paths remain independent. A portion of product's FPGA is dedicated to extracting the clock data embedded in the incoming AES signal. As is true for nearly every element of our products, the circuitry is bespoke, rather than an off-the-shelf chip. The FPGA is used to perform other tasks, such retrieving and routing the audio, processing it, and so on.

The programmability of our platforms gives us a tremendous amount of flexibility in terms of how we facilitate nearly every function of the product. One example is our dual AES design—we can run the signal, have a separate Master Clock input, employ the DAC as the Master Clock for the whole audio system, tolerate different lengths of cables in Dual AES, and deal with phase offset between clock and audio. All of this can be done without adding latency to the audio, meaning it can still properly integrate with video. We are also able to hide commands embedded in the non-audio bits of AES, which allows, for example, the Vivaldi DAC (a non-network equipped product) to be controlled by the dCS Mosaic Control app.




*This diagram shows an example of how a digital source (a Rossini Transport), a DAC (the Bartók Headphone DAC) and a master clock (the Rossini Clock) work together. The overall performance of the system is reliant on each of these stages performing correctly – each oscillator, PLL and output stage needs to operate at a high level to achieve optimum performance.*

# Clock Dither

The setting for Dither can be found on the dCS Rossini and Vivaldi Clocks. Dither is commonly found in digital audio, where it is used to expose dynamic resolution below that of the least significant bit. In the aforementioned Clocks however, the dither is applied to the time domain instead of the amplitude domain.

PLLs exhibit what is known as a "dead band" in their phase detectors. When the input and output frequencies are close to being synchronised, they lose sensitivity. The PLL then drifts until the difference in frequency is large enough to cause the phase detector to once again become active and drive the PLL back towards being synchronised.

This is where the dither comes in: Perhaps counter-intuitively, if very small, random variations in the timing of the clock signal edge are applied when the phase error is very low, it gives the PLL something to latch on to and correct. It pushes the phase error slightly back into the area where the phase detector can correct well. The dither is then filtered out in the PLL before it outputs the final clock signal. In practical listening, this is a good trade-off and actually improves system performance. In essence, the dither setting on the Rossini Clock allows the Bartók DAC's clock to be very accurate even when the PLL is working in a less sensitive, low phase error area.

# Asynchronous Sources – USB & Network Audio

Audio sent over an asynchronous format (such as streaming to a smartphone via Spotify, playing content from a NAS via Roon, or playing music from a computer via USB) is, to an extent, the exception to the rules stated in the previous posts, in so much as jitter is not a factor for the audio data, at least until it reaches the endpoint and is converted back to the relevant format (such as PCM or DSD).

With network audio, the interface that is used to send audio data over a network is called TCP (Transfer Communication Protocol). The data which is to be transmitted from one place to another – in this case a piece of music – is split up into multiple packets. These packets contain not only the data itself (the "payload"), but tags on where it has come from, where it is going, how many packets it is part of and how these packets should be reassembled to get the original data back unchanged.

Take, for example, a track from Qobuz being streamed to a dCS Bartók DAC. If a packet of data is lost or compromised, according to the TCP interface, the Bartók can simply request that packet again. When all the correct packets have been received properly by the Bartók, they are unpacked back to the correct data format (PCM, for example) and buffered before being fed to the DAC. This stage, the unpacking and buffering, effectively removes any timing link between the TCP packets and the resultant audio signal. (Read that sentence again, as it's very important.)

Once the data has been buffered in the Bartók, the factors discussed above become relevant again. The data is now being directly dictated by the Bartók's clock and as such, jitter becomes a factor. The accuracy of the Bartók's clock then controls when the DAC converts the samples back to analogue voltages, so has a direct impact on audio quality. Until it reaches that point, however, jitter is simply not a factor from an audio perspective.

Asynchronous USB audio works in a similar way. There is no timing link whatsoever between the source, such as a computer, and the endpoint such as a Bartók. It does not matter if the bits are not perfectly spaced as a clean square wave while the USB data is being transferred. Provided the bits are received by the Bartók correctly (a 1 isn't misread as a 0, for example), the timing is largely irrelevant. This is because, as with network audio, the data is buffered before being fed to the DAC. It is not until this point that timing becomes a factor, as at this point, it has been converted back from USB format to digital audio (eg PCM or DSD).

*dCS* Insights & Innovation

# Clocking in digital audio

**dcsaudio.com** | **@dCSonlythemusic** | **info@dcsaudio.com**

Data Conversion Systems Ltd.
Unit 1, Buckingway Business Park, Anderson Road,
Swavesey, Cambridge, CB24 4AE, UK

dCS
ONLY THE MUSIC