

Nuendo - Synchronization

Background

What is synchronization?

Synchronization is said to exist when you make two pieces of equipment agree on time or tempo. You can establish synchronization between Nuendo and a number of other types of devices, including tape recorders and video decks, but also MIDI devices that “play back”, such as other sequencers, drum machines, “workstation sequencers” etc. When you set up a synchronization system you must decide which unit is the *master*. All other devices are then *slaved* to this unit, which means they will adjust their playback speed to the master’s.

Nuendo as a slave

When a synchronization signal is coming in to Nuendo, from another device (such as a tape recorder, video recorder etc.), this device is the master and Nuendo is the slave. Nuendo will adjust its playback to the other device.

Nuendo as a master

When you set up Nuendo to transmit synchronization information to other devices, Nuendo is the master and the other devices are the slaves; they will adjust their playback to Nuendo.

Nuendo – both master and slave

Nuendo is a very capable synchronization device. It can operate as both a master and a slave at the same time. For example, Nuendo might be slaved to a tape recorder transmitting timecode, while at the same time transmitting MIDI Clock to a drum machine, acting as a master for that.

The VST System Link feature (with which you can synchronize separate computers running Nuendo or Cubase SX) is described in a separate chapter. See page 605.

Timecode, MIDI clock and word clock

Basically there are three types of synchronization signals for audio, Timecode, MIDI clock and word clock:

Timecode (SMPTE, EBU, MTC, VITC etc.)

Timecode appears in a number of guises. No matter which “format” it has, it always supplies a “clock on the wall” type of synchronization, that is, a synchronization related to hours, minutes, seconds and two smaller units called “frames” and “sub-frames”.

- LTC (SMPTE, EBU) is the audio version of Timecode. This means that it can be recorded on the audio track of an audio or video recorder.
- VITC is the video format Timecode that is it is stored in the actual video image.
- MTC is the MIDI version of Timecode, transmitted in MIDI cables.
- ADAT sync (Alesis) is only used with the ASIO Positioning Protocol, see page 585.
- Sony 9-pin is a standard that uses serial (RS-422) communication. Sony 9-Pin contains Timecode as well as machine control messages (see page 598).
For the ASIO Positioning Protocol, other high precision Timecode formats may also be supported.

Format recommendations for Timecode – without ASIO Positioning Protocol

- When synchronizing your system to external Timecode, via a synchronizer, the most common Timecode format is MTC. Contrary to some reports you might have heard, MTC delivers good precision for external sync. This is due to the fact that the operating system can “time stamp” incoming MIDI messages, which increases precision.
- Sony 9-Pin is not recommended for external sync and should only be used when no other option is available. However, you might of course use 9-Pin for machine control, see page 587.

Format recommendations for Timecode – with ASIO Positioning Protocol

- LTC and VITC are the formats with the highest precision and are recommended when available.
- MTC is the next best option and probably the most common choice, since few audio hardware solutions have built in LTC or VITC readers. However, LTC and VITC offer even higher precision when available.
- The choice with the least precision is Sony 9-Pin.

MIDI Clock

MIDI Clock is a tempo based type of synchronization signal that is related to the number of “beats per minute”. MIDI Clock signals are suitable for synchronizing two devices that agree on tempo, such as for example Nuendo and a drum machine.

MIDI Clock is not suitable as a master sync source for an application like Nuendo. Therefore Nuendo will transmit MIDI Clock signals to other devices, but it will not receive MIDI Clock.

Word Clock

Word clock is basically a replacement for the sample rate clock in for example an audio card. Word clock hence runs at the same rate as the sample rate in the audio, 44.1kHz, 48kHz etc. Word clock does not contain any position information, it is only a “simple” signal for clocking the audio at its sample rate. Word clock comes in many formats, analog on coaxial cable, digital as part of an S/PDIF, AES/EBU or ADAT audio signal, etc.

Synchronizing the transport vs. synchronizing audio

How timing is handled in a non-synchronized system

Let's first look at the situation where Nuendo is not synchronized to any external source. Any digital playback system has an internal clock that ultimately affects the playback speed and stability, and PC audio hardware is no exception. This clock is *extremely* stable. When Nuendo is playing back with no external synchronization, all playback is internally synchronized to the internal digital audio clock, to ensure synchronization between digital audio and MIDI.

Synchronizing Nuendo's playback

Let's assume now that we use external Timecode synchronization, with Nuendo. For example, we might synchronize playback to a tape recorder. Timecode coming from an analog tape recorder will always vary slightly in speed. Different Timecode generators and different tape recorders will also supply Timecode with slight differences in speed. In addition, the shuttling of tape mechanisms due to overdubs and re-recordings can cause the physical tape to wear and stretch, which affects the speed of the Timecode.

If you set up Nuendo to sync to incoming Timecode, it will vary its overall playback speed to compensate for such fluctuations in the speed of the Timecode – that's the whole purpose of synchronization.

What happens with the digital audio?

The fact that Nuendo's playback is synchronized to the Timecode does not affect the playback of the digital audio. It still relies on the perfectly stable, built-in clock in the audio hardware. As you may understand by now, problems will appear when the perfectly stable digital audio gets related to the slightly varying speed of a system synchronized to Timecode. The playback timing of each event will *not* be in total accordance with the tape or the MIDI playback, since the playback speed of the audio is determined by the digital audio hardware's built-in clock.

Resolving to word clock

The solution to this problem is to use one external clock for all components in the system. One master clock is used to derive whatever type of clock signal each component in the system needs. For example, something called a house clock can be used to generate sample rate clocks for the digital audio hardware and Timecode for Nuendo. This ensures all components in the system use the same reference source for their timing. Synchronizing digital audio to external clocks running at sample rate is often called "resolving" or "synchronizing to word clock". If you aim to perform synchronization to external signals, we strongly recommend that you obtain proper synchronization equipment. This encompasses:

- An audio card that can be slaved to external word clock.
- A synchronizer that can read Timecode (and possibly house clock) and generate the required sync signals from that, such as the Steinberg TimeLock Pro.
or...
- An audio system with complete built-in synchronization possibilities, preferably supporting the ASIO Positioning Protocol.

Using Timecode without word clock

Of course, it is possible to set up a synchronization system where you lock Nuendo to Timecode without using word clock. However, please note that the timing of audio vs. MIDI cannot be guaranteed and that fluctuations in speed in the incoming Timecode will not affect the playback of audio events. This means that synchronizing to Timecode may work in the following situations:

- When the Timecode was originally generated by the audio card itself.
- When the source providing the Timecode is extremely stable (such as a digital video system, a digital tape recorder or another computer).
- When you remain synchronized to that same stable source throughout the entire process, both while recording and playing back audio.

About the ASIO Positioning Protocol (APP)

ASIO Positioning Protocol requires audio hardware with specific ASIO drivers. As of this writing, APP is not available for audio hardware with non-ASIO drivers under Mac OS X.

The ASIO Positioning Protocol is a technology that expands on the type of sync described above and makes *sample accurate positioning* possible.

When transferring audio digitally between devices, it is important that synchronization using word clock and Timecode is *completely* correlated. If not, the audio will not be recorded at the exact intended (sample accurate) position, which can cause various types of problems, such as inaccurately positioned audio material, clicks and pops etc. A typical situation is when transferring material from a digital multi-track tape recorder to Nuendo (for editing) and then back again. If you do not have sample accurate synchronization set up, you cannot be sure that the material will appear in its exact original position, when transferred back to the tape recorder.

In order to take advantage of the ASIO Positioning Protocol, your audio hardware must be suitably equipped and the functionality must be included in the ASIO driver for the hardware. An example of a system for doing sample accurate transfers would be transferring audio tracks from an Alesis ADAT to Nuendo. Here the ADAT will be the sync master (though it doesn't necessarily have to be). It provides both the digital audio (with an inherent word clock) and positional information (Timecode) via its ADAT sync protocol. The master clock is generated by the ADAT itself.

Hardware and software requirements for APP

- Your computer audio hardware (in our example above this would be an ADAT card in your computer) must support all the functionality required for the ASIO Positioning Protocol. That is, it must be able to read the digital audio and the corresponding positional information from the external device.
- There must be an ASIO 2.0 driver for the audio hardware.
- For resolving to external Timecode, the audio hardware must have a Timecode reader/writer on-board.
- For information about which audio hardware models currently support APP, see the Steinberg web site (www.steinberg.net).

The ASIO Positioning Protocol exploits the specific advantage of having an audio card that has a built-in Timecode reader. With such a card and the ASIO Positioning Protocol, you can achieve constant sample accurate synchronization between the audio source and Nuendo.

Machine control

Nuendo can control external tape transports and similar devices via MIDI Machine Control or Sony 9-Pin (more details about each format follow below). This allows you to operate an external tape transport from Nuendo's Transport panel. That is, Nuendo can make the tape recorder locate to a certain position, start, stop, rewind etc.

About sync and machine control

Controlling tape transports is a two-way process:

- Nuendo sends out machine control commands to the tape recorder, asking it to locate to a certain position and activate playback etc.
- The tape recorder locates to the requested position, starts and delivers Timecode back to Nuendo, to which Nuendo is synchronized.

Even though it appears as if Nuendo is controlling the tape recorder completely, it is important to remember that in this setup, Nuendo is still being synchronized to the external tape transport, not vice versa.

Also note that the two processes of sync and machine control are completely separated, in terms of protocols used. You can for example synchronize to MTC while sending out transport commands via MMC.

MIDI Machine Control (MMC)

This is a standard MIDI protocol for controlling tape transports. There are a number of tape recorders and hard disk recording systems on the market that support this protocol. Nuendo implements three MMC commands: Start, Stop and Locate.

Sony 9-Pin

This is a standard protocol established by Sony, for controlling audio and video tape transports. It is implemented via regular serial port communication (RS-422). The correct cabling varies between computer platforms and models. Please refer to page 597 to find out more about the correct setup for your computer.

- If your Sony 9-pin device support RS-232 you need a "null modem cable".
- The Sony 9-pin device most likely needs to be switched from local control to remote control, from its front panel. If the device is not set to the correct mode, Nuendo will inform you.
- Normally you should lock Nuendo to Timecode from the Sony 9-Pin device, not another source.
- Nuendo implements Start, Stop, Record and Locate commands for 9-Pin devices, along with some special features, described on page 601.