Synchronizing Delta1010LT sound cards

By Bohdan Raczynski
Bodzio Software Pty. Ltd.

Sound cards have built-in clocks, which provides time reference for their internal software to runs accordingly to their individual clocks. Even though these clocks are crystal-controlled, it is unlikely, that they run on exactly the same frequency. Even small difference, which is perfectly within manufacturing tolerance, will cause the data streams coming out from the two sound cards to drift apart in time. Eventually, clicks and cracking sound will become evident. Running two sound cards on their internal and separate clocks is really not recommended.

Accordingly to M-Audio Delta1010LT internet comments posted on their User’s Forum, “The newest drivers can do multi-card sync, but it is officially not supported to do so…..There are two ways to synchronize the two cards, S/PDIF or Word Clock. It's up to you to determine which one is better/easier for you. 75ohm cable should be used either way….”.

http://forums.m-audio.com/showthread.php?28105-Delta-Multi-Card-Direct-Test

Avid (formerly M-Audio) is actually encouraging reporting issues with multi-cards setups. Well, this is actually quite encouraging.

Indeed, installing the second Delta1010LT on the test computer, caused Windows7 to accept the card right away, and the second card become visible in the Delta1010LT control panel and in Windows 7 Sound System. Exactly as you would expect. Then, the Word Clock output from “master” card (the originally installed) needs to be connected to the Word Clock input of the “slave” card (newly installed). The synch source on the slave card has to be set to “external Word Clock”. The “Synchronized” green indicator is ON, but may blink red every 5-6 second. This can be disregarded, accordingly to M-Audio.

From now on, the two delta1010LT cards are synchronized, and will not drift apart in time. This is a great progress. However, when you check the IRQ assignments in Windows Device Manager, the originally installed Delta1010LT will be serviced by IRQ = 16, and the later installed Delta1010LT will be serviced by IRQ = 17. Commonly, the IRQs are polled within approximately 10ms. Now, even though the cards are synchronized, the polling causes significant problem. It would be much better, if the two Delta1010LT sound cards appeared to the Device Manager as one, and be serviced by single IRQ = 16. Unfortunately, the driver is not designed this way.

Imagine, that your system design calls for woofer and tweeter in the same enclosure to be powered by different sound cards. This would likely result, in woofer and tweeter receiving their corresponding signals separated by 10ms apart. Generally, is this a problem?.

Perhaps you could arrange your system such a way, that all drivers constituting all front loudspeakers would be powered from one sound card. For instance, you could arrange for FL = 3way, C = 2way and FR = 3way systems, that would require 8 (fully synchronized and time aligned) outputs from the first Delta1010LT. Then you could assign RL = 3way, Sub = 2x1way, and RR = 3way, that would require 8 (again, fully synchronized and time aligned) outputs from the second Delta1010LT. This could work, but the rear speakers and subwoofer would be delayed by around 10ms. This translates to an “artificial” distance of 3.4 meters. For some users, this may not be a problem at all. So, this “do nothing” approach may still be an option.

Solution 1

Permanently delay all drivers in one sound card by 10ms. This can be done in UltimateEqualizer V3 (UE3). This brings the front-to-rear time alignment within +/-1ms, or +/-48 sample times at 48kHz. This could be more acceptable solution and certainly, would not be audible.

Solution 2

A number of measurements designed to record differential latency between the two, synchronized Delta1010LT sound cards, have been performed. The results indicate, that the differential latency varies from 8-12ms and looks basically like a random number within this numerical brackets. However, this latency appears to be stable for each session – it does not change from one data frame to another. Every time you press “Play” button on the UE3 control panel, the audio data blocks within the two sound cards will be processed with a latency of 8-12ms, but the latency will be fixed for each session within one sample time accuracy.

So, if one could automatically measure latency of each session, and delay the “faster” sound card by the exact number of samples?. This would, in theory, align the data streams from both sound cards within one sample-time accuracy.

Dynamic Synchronization scheme for two Delta1010LT sound cards

The essence of the scheme is to align data streams fed to the output D/A converters of both sound cards to within single sample time. Since the D/A are now clocked synchronously via the Word Clock mechanism, the outputs should be in synch.

To accomplish this, the differential latency of each audio session between the two Delta1010LT sound cards needs to be measured, and the faster sound card needs to be delayed by this exact number.

The scheme employs short MLS burst generated at Output 1 of the Delta1010LT N#1, and the same MLS burst is generated at Output 1 of the Delta1010LT N#2. The bursts are software triggered at the same time, and are being fed back to the unused inputs of The Delta1010LT N#1 – these are Input 7 and Input 8. This whole scenario is identical to performing “loop-test” on the MSL system.

After converting the MLS bursts to Impulse Responses, one can clearly see, that their sharp peaks are delayed in time by several hundred sample times. The distance between the IR peaks is the differential latency figure we are looking for.

In the final step, the faster (typically the first Delta1010LT) is delayed by the differential latency and from now onwards, the two sound card outputs with MLS bursts are released back to the system as normal audio outputs.

The MLS burst is 16k long, which for the sampling frequency of 48kHz, translates to 340ms length of time. The amplitude of the MLS burst is only 10% of the maximum level, so it will not harm a tweeter connected to the same output line. As additional precaution, the MLS can be “colored” type, so the low-frequency content is basically HP-filtered.

So in summary, every time when you press the “Play” button on UE3 control panel, the system emits 340ms of quiet noise on two outputs, and then plays as normal. Sound cards are synchronized and time aligned to within one sample time.

This can be further improved. If you do not use all 16 outputs, you can perhaps nominate one output from each soundcard as the MLS outputs for the latency measurements. This way, you can take the measurements completely out of audio channels, and you’ll never even know, that the system performs them.

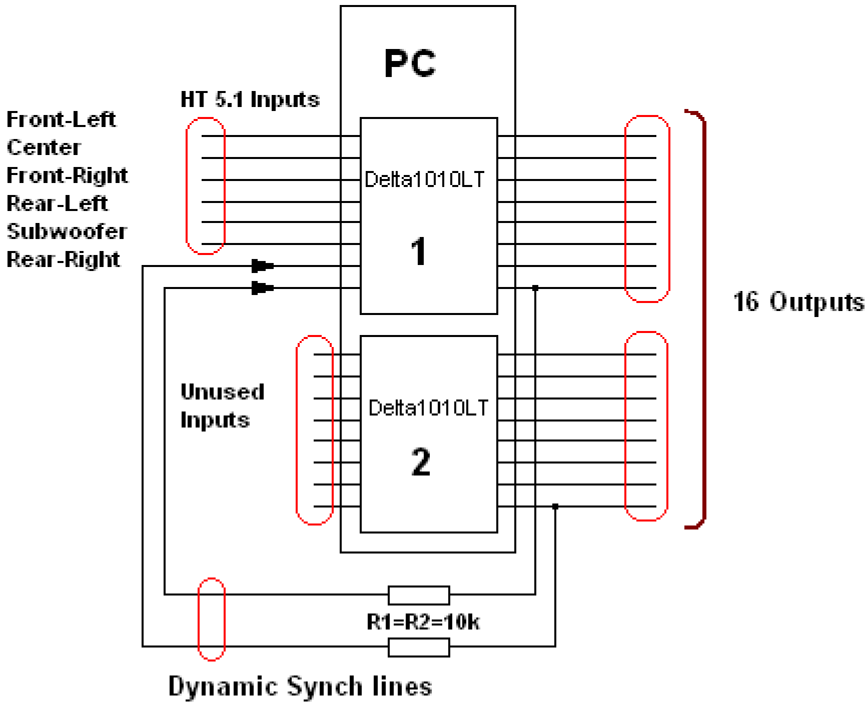


Figure 1. Dynamic Synchronization scheme for 2 x Delta1010LT.

How accurate is the “Dynamic Synch” synchronization method?

When the computer is turned on, the sound card driver will synchronize both cards connected via word clock mechanism, and UE3 will align them in time for each audio playback session. The synch should be within-one sample time. For 48kHz sampling rate, this would equate to 20.833microseconds. The maximum phase error resulting from such alignment is illustrated on Figure 2.

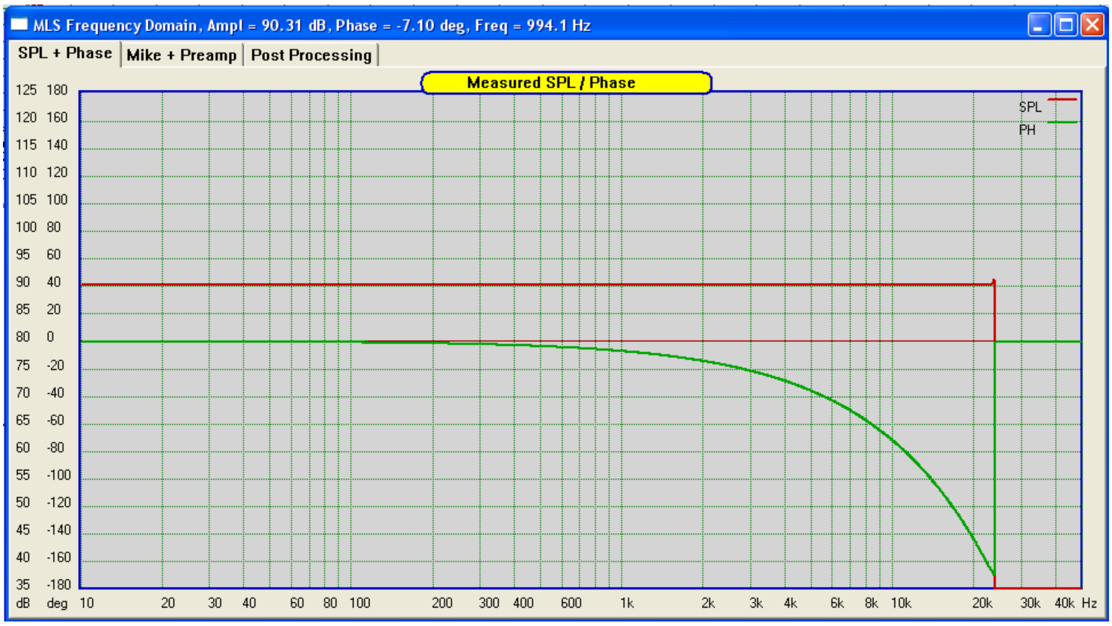


Figure 2. Maximum phase error (green curve) between synchronized cards.

WDM buffer size of 256 and 1024 works well on the test computer. Operating system is Windows 7/64bit. Delta1010LT driver is: 5.10.0.5074.

Please select ASIO/WDM buffer size as 256 samples.

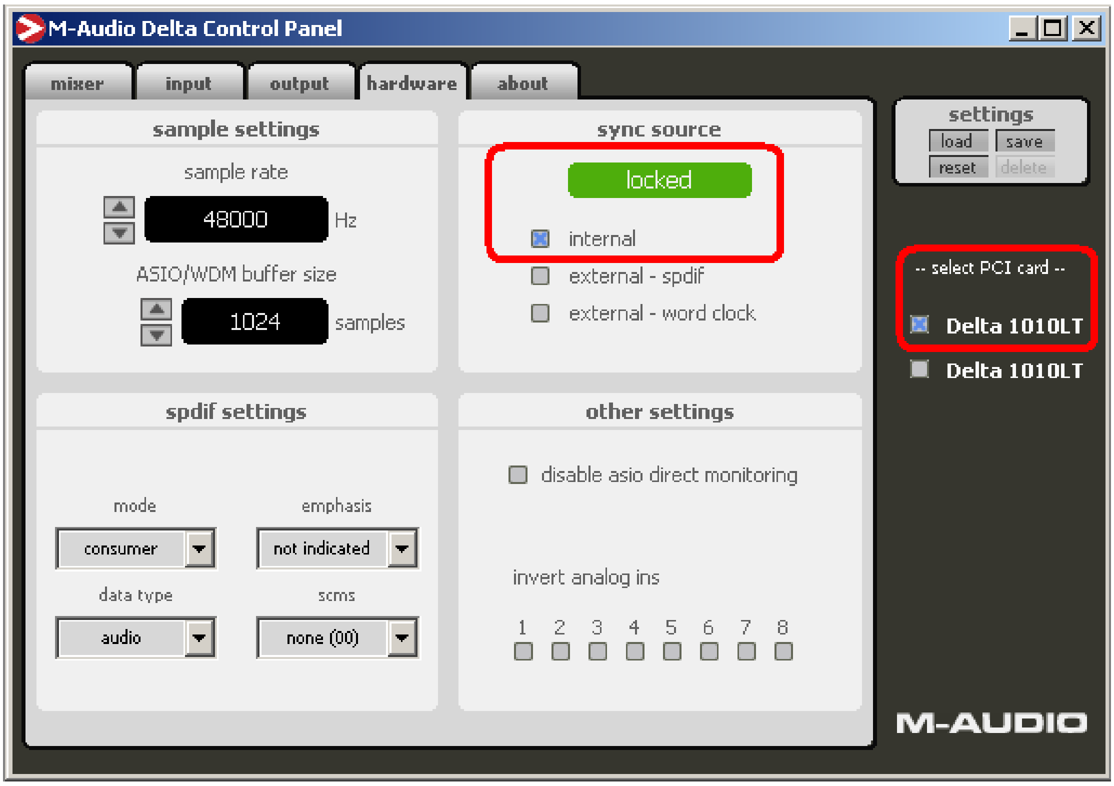


Figure 4. “master” delta1010LT sound card settings.

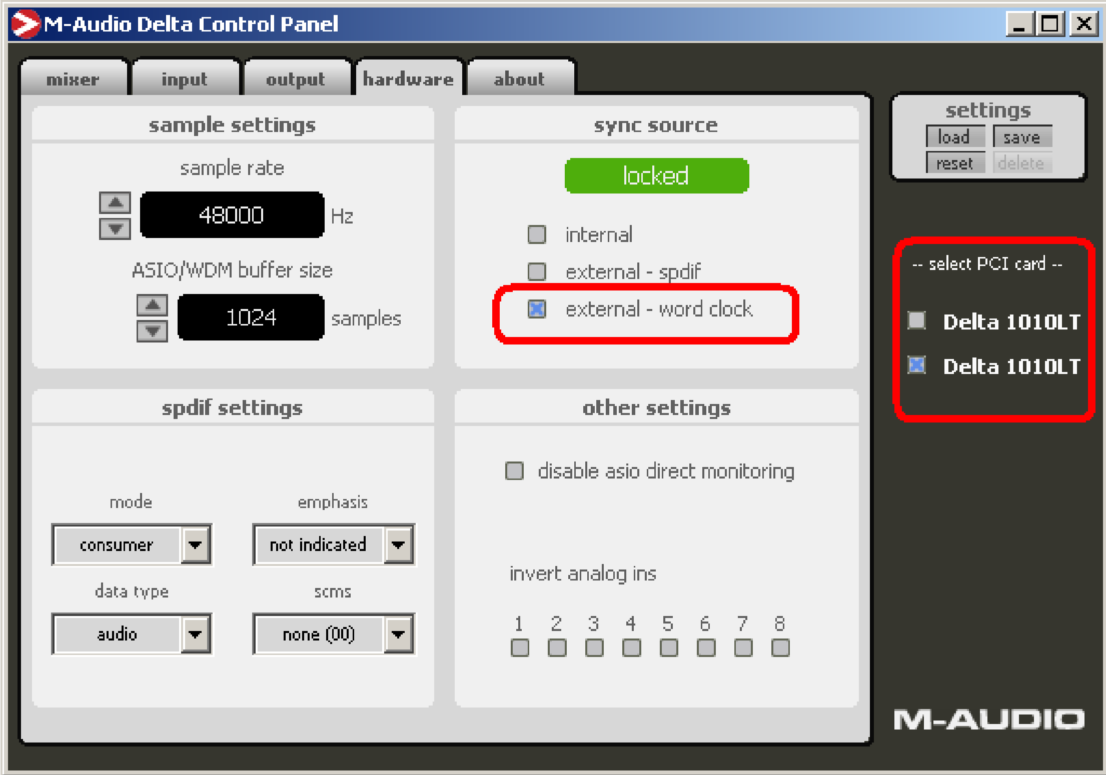


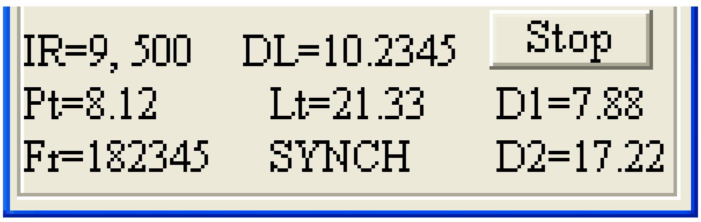
Figure 5. “slave” Delta1010LT sound card settings.

There are some advantages of using two Delta1010LT cards, as opposed to more elaborate Trace8 + TraceOut A16 combination:

1. Lower cost – two Delta1010LT cards ($460 in total) are still cheaper than one TraceOut A16 ($480).
2. Windows7/64 + Windows7/32 compatibility. Trace cards are only Windows7/32bit compatible for now.
3. Delta1010LT puts out 10Vpp, and TraceOutA16 puts out 5Vpp. This results in +6dB better S/N ratio for no extra effort.

On the other hand, using synchronized Trace8+TraceOutA16 will offer the user 8in/24out, audio channels. This would be the only way to create 24-output device with 2xPCI slots on the PC motherboard.

The following parameters are measured and/or continuously monitored for two synchronized sound cards:



1. **IR = 9, 500** Location of the IR peaks on both sound cards. In this example, the IR peak location of the "faster" sound card is 9 sample times, and the location of the "lagging" sound card is 500 sample times. These numbers will change for every audio session triggered by the "Play" button.
2. **DL = 10.2345 (Delay)** Calculated delay (in milliseconds) necessary to bring the two sound cards into alignment. This number will change for every audio session triggered by the "Play" button.
3. **Pt = 8.12 (Processing time)**Time taken (in milliseconds) to process 16-output partitioned convolution on one block of input data. This number will fluctuate slightly during the operation of the DSP engine.
4. **Lt = 21.33 (Lapsed time)** Lapsed time (in milliseconds) taken by WASAPI audio system to deliver consecutive blocks of audio data. This number should be close to 21.334ms. This number will fluctuate slightly during the operation of the DSP engine.
5. **Fr = 182345** Running tally of processed frames.
6. **D1 = 7.88** Figure of merit for faster sound card. This number should be within 4-9 units. This number will fluctuate slightly during the operation of the DSP engine.
7. **D2 = 17.22** Figure of merit for lagging sound card. This number should be within 12-19 units. This number will fluctuate slightly during the operation of the DSP engine.
8. **SYNCH** Indicator, that the sound cards are synchronized.

Conclusions

Synchronizing two Delta1010LT sound cards can be accomplished within one sample-time, or 20.833usec for 48kHz sampling rate. This would pave the way to incorporating two cards in a large 5.1 Home Theatre system, consisting of up to 16 individual drivers.

For an uncompromising performance, it is advisable, to group drivers comprising individual loudspeaker boxes to the same sound card. For instance, FL = 4way, FR = 4way would be assigned to one sound card, and C = 3way, RL = 2way, RR = 2way and Sub = 1way, would be assigned to the second sound card, totaling 16 outputs. Many other combinations are possible as well.

It should be noted, that the Dynamic Synchronization scheme described above is capable of synchronizing the two sound cards perfectly – that is with no phase error between the two sound cards at all. This would have happened, had the driver been capable of clocking D/A converters on both sound cards precisely at the same time.

However, the limit of final accuracy of the scheme is determined by the Word Clock (or SPDIF) synchronization accuracy, provided by the sound card driver. And this is up to one sample time.

Solution 3

Strategically insert into the program, a few high-resolution performance counters, based on two functions: (1) QueryPerformanceFrequency(), which retrieves the frequency of the high-resolution performance counter in counts per second and (2) QueryPerformanceCounter( ), which retrieves the current value of the high-resolution performance counter.

Using this approach it is possible to measure with several micro-second accuracy, the timing of audio data blocks supplied to both sound cards by the Windows 7 audio engine. If the jitter of the audio engine is low, the method should yield far better synchronization accuracy then simply supplying a fixed delay for the advancing card. In addition, this method can be fully automated and does not require user’s intervention, but most importantly, it will calculate and adjust the delay for each audio session individually. In this respect, this method is similar to the Dynamic Synchronization MLS method.

Accuracy of this method is within a few samples for 48kHz sampling rate. Shown below is the output of two sound cards captured on a digital CRO, while connecting sine waves to both inputs at the same time. It is observable, that with 1ms/div time scale, there is no discernible difference between the outputs.