Creating a DIY non oversampling DAC with PCM1704.

Non oversampling DACs are by many people (includung me) regarded as better sounding than conventional oversampling systems. The easiest way to enjoy that kind of sound is to build a DAC with the "outdated" TDA1543/TDA1541 DAC ICs or modify an old Philips/Marantz CD player.

But why not building a nonos DAC with higher quality converters like PCM1704, PCM1702 or AD1862? Well, you need to convert serial DATA to parallel DATA which requires some glue logic. Further glue logic is needed to align 16bit DATA with the 20bit or 24bit inputs of these DAC ICs.

While there are many well documented TDA based nonos projects available on the www, almost nothing can be found about building a nonos DAC with more recent Burr Brown / Analog Device ICs. That's why I'll try to give some advice and share some of my experience in building such a DAC. If there wouldn't be this glue logic issue, I guess more DIYer would build nonos DACs with Burr Brown chips.

This paper does not provide a complete schematic for a DAC. Not even a schematic for interfacing decoder A with DAC X. But it should support and encourage you in designing such an interface for your own non oversampling DAC.

First of all it's important to decide whether you're going for a separate DAC (fed with SPDIF) or a one box CD player. The one box solution (without SPDIF) is definitely to be preferred. It's a little better sounding, cheaper and more practical (imho). Of course you can add an input reciever to your CD player and use it also as an external DAC.

There are basically two kinds of CD players. The Sony and the Philips based engines. Many decoders (SAA7210/CXD25XX/...) used in those players put out a non oversampled (1fs) true 16bit signal as stored on the CD. On the most common input recievers you have access to different output formats, including Sony format, Philips I2S and usually some more.

All decoders and input recievers put out the following three signals which need to be connected to the DAC ICs:

BCK:
System tact, let's the converters know, at which time to read the DATA bits.

LRCK:
Let's the converter know which word belongs to left and right channel.

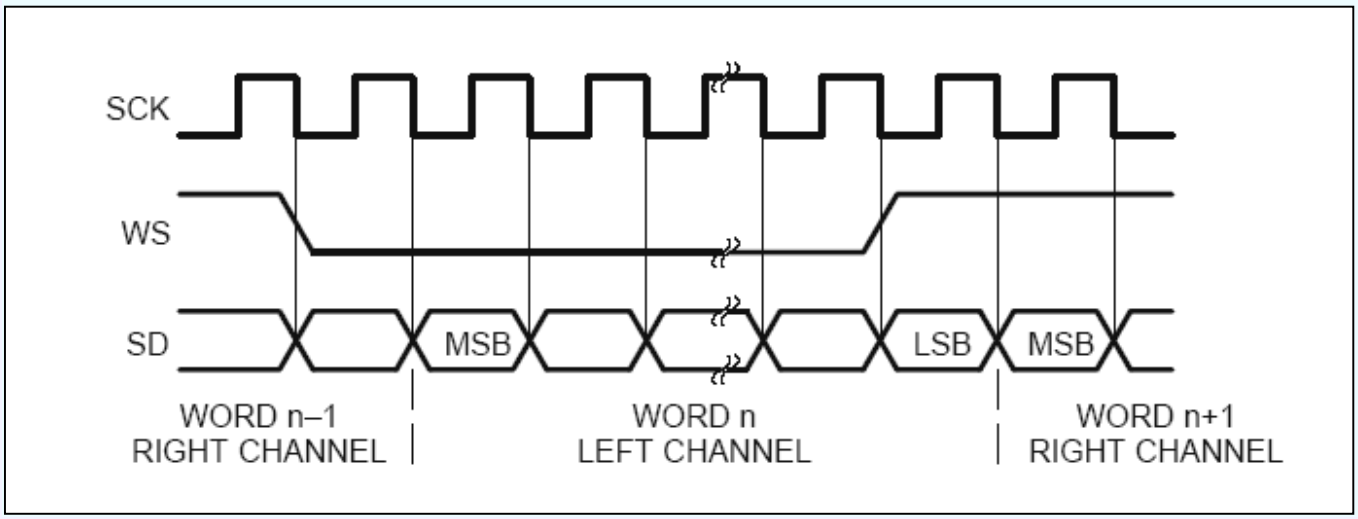
DATA:
Transmits two 16bit words at a sampling rate of 44.1kHz. Transmission starts with the left word.

Philips I2S:

BCK frequency is 2,8224MHz (44,1kHz x 64). 44.1kHz is multiplied by 64 because there are two words per sample transmitted in a word frame length of 32 BCK cycles.

LRCK frequency is 44.1kHz (1fs). LRCK being low indicates the left word, LRCK being high indicates the right word.

DATA is encoded in binary twos complement (BTC) and transmitted the most significant bit (MSB) being first. The MSB is aligned with the second rising edge of BCK after LRCK has changed. DATA words are left justified with delay of one BCK cycle .



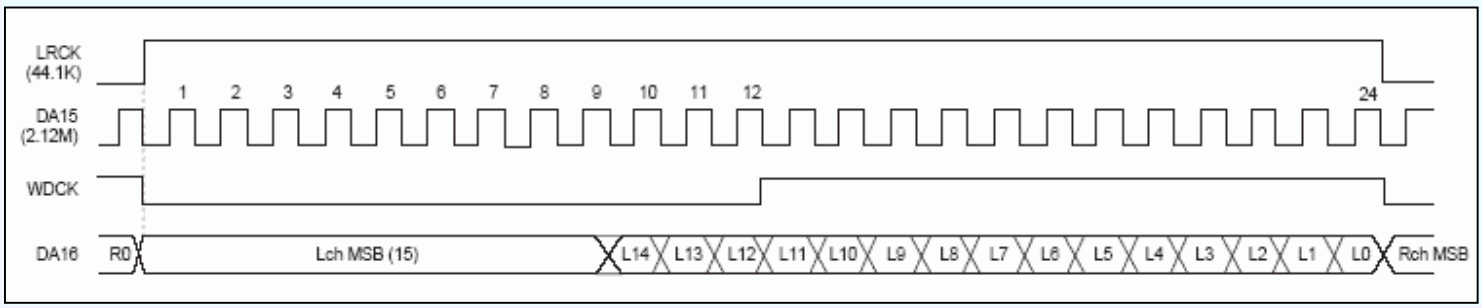
Sony format:

BCK frequency is 2,1168MHz (44,1kHz x 48). 44.1kHz is multiplied by 48 because there are two words per sample transmitted in a word frame length of 24 BCK cycles.

LRCK frequency is 44.1kHz (1fs). LRCK being low indicates the right word, LRCK being high indicates the left word.

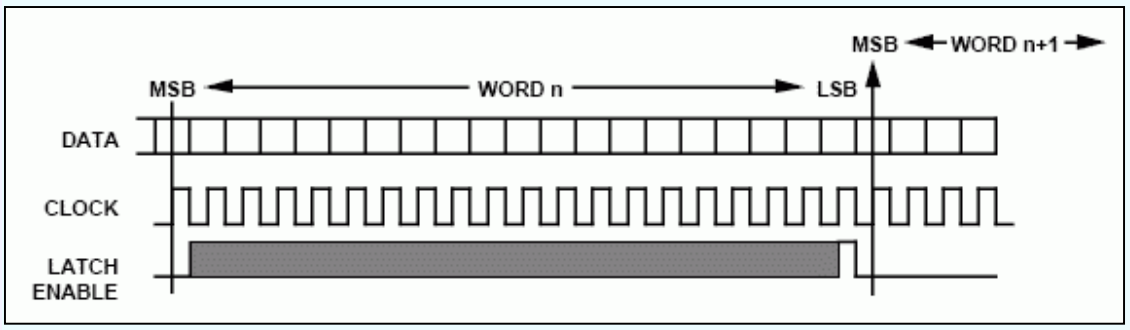
DATA is encoded in BTC and transmitted the MSB being first. The MSB is aligned with the ninth rising edge of BCK after LRCK has changed. DATA words are right justified.

Sony decoders put out the MSB from the first till the ninth rise of the BCK. This will lead to some distortion at low signal levels, when DATA is shifted and fed to a DAC with more than 16bit input. With a 20bit DAC the MSB of the next word will be regarded as the four smallest bits of the present word. CD players using Philips decoders are preferred.

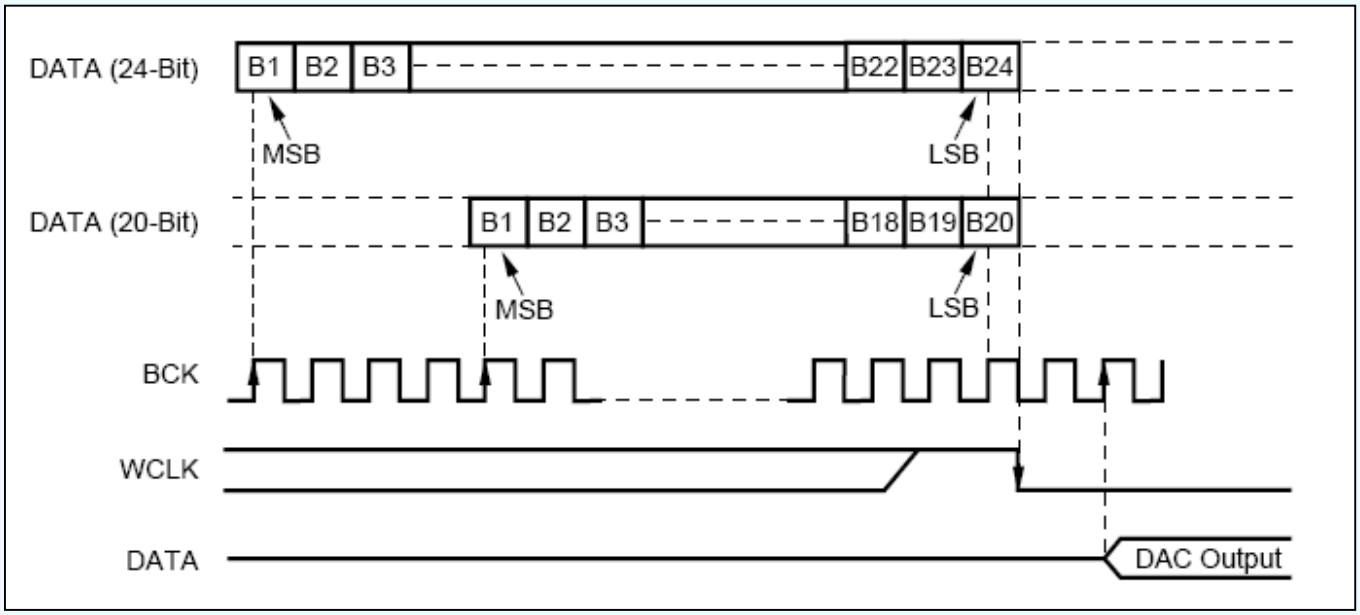


The inputs of the Analog Device and Burr Brown ICs are very similar.

DATA is always right justified. The least significant bit (LSB) is clocked in at the last rising edge of BCK before latch enable (LE) goes down.



AD1862



PCM1704

Now that we want to have left and right word in parallel, we always have to delay the left word by 24 or 32 BCK cycles, depending on the word frame lenght of the decoder.

Next step is to bring the MSB into the correct relation to LE:

When using a Philips decoder, we have to delay serial DATA by another 7 or 11 BCK cycles (24bit or 20bit DAC IC). LRCK can directly be used as LE.

When using a Sony decoder, we have to delay LRCK instead of DATA. LRCK must be delayed by 8 or 4 BCK cycles (24bit or 20bit DAC IC). LRCK must be inverted and can then be used as LE.
Solution II: Instead of shifting LRCK, serial DATA can be delayed by another 20 or 16 BCK cycles (20bit or 24bit DAC IC). By doing so, LRCK can directly be used as LE.

There are of course further possibilities. Other formats from different input receivers or combinations with 18bit converters. Just look closely at the datasheets and figure out what has to be shifted and what must be inverted. If you've done that you're almost there.

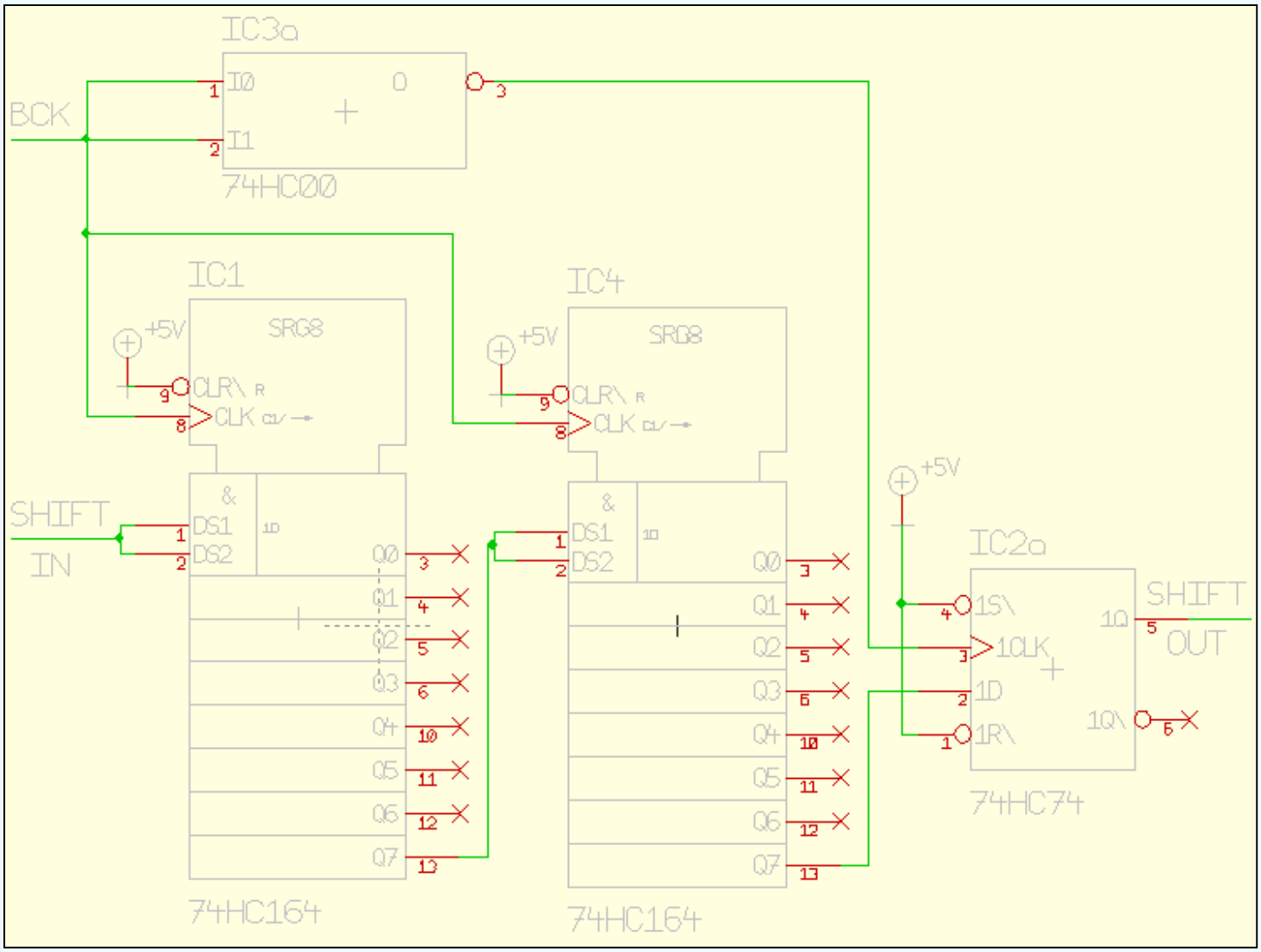
The question now is how to implement this. To create a shifter, all you need are three different logic modules:

Inverter: 74HC00/74HC04/...

Shiftregister: 74HC164

D-Type Flip Flop: 74HC74

The datasheets will help you understand how the explanatory circuit below works. If you've got that, just expand this circuit to get the right amount of shifting and correct LE.



Q7 (IC1) puts out the signal present on "SHIFT IN" delayed by 7.5 BCK cycles.
Q7 (IC4) puts out the signal present on its inputs (DS1/DS2) delayed by 8 BCK cycles.
The inverted clocked 74HC74 adds another delay of 0.5 BCK cycles.

A comlete example of how to combine an input reciever with an Analog Device 18bit DAC IC can be found here.

You will never need more than 8 logic IC's to combine any decoder/receiver with any DAC IC. It could also be done in simpler ways, e.g. "stopped clock operation" like explained in Analog Device application note 207 (AN207). Or you could do it without serial to parallel conversion similar as shown here (search for "2xPCM1702" and see post #13).

Unfrotunately it never worked with stopped clock operation. So I tried the simple version without serial/parallel conversion. I wasn't able to hear the phase shifts (see here for explanation) between left and right channel. But I wanted to have a flawless DAC, so I've implemented serial/parallel conversion and I'm very happy with it.

Up till now I've done nonos with PCM1704, PCM1702 and AD1862. All are built as one box CD players based on Marantz, Denon and Philips. All of them sound superior to my tuned (also non oversampling) Philips CD650 with TDA1541/S2 or my DDDAC1543.

Another nice idea is to build 24bit/96kHz nonos converter for Music DVD's. When usig Philips I2S signals (or another left justified format), you can feed anything from 16bit/16kHz to 24bit/192kHz to your converter through the same shifting interface.

Anyway, I'd say it's highly recommended to do nonos with a really good DAC IC ;-)

Download this page as PDF.