INTERFACING THE VS1053 AND VS1063 TO DACS AND SRCS

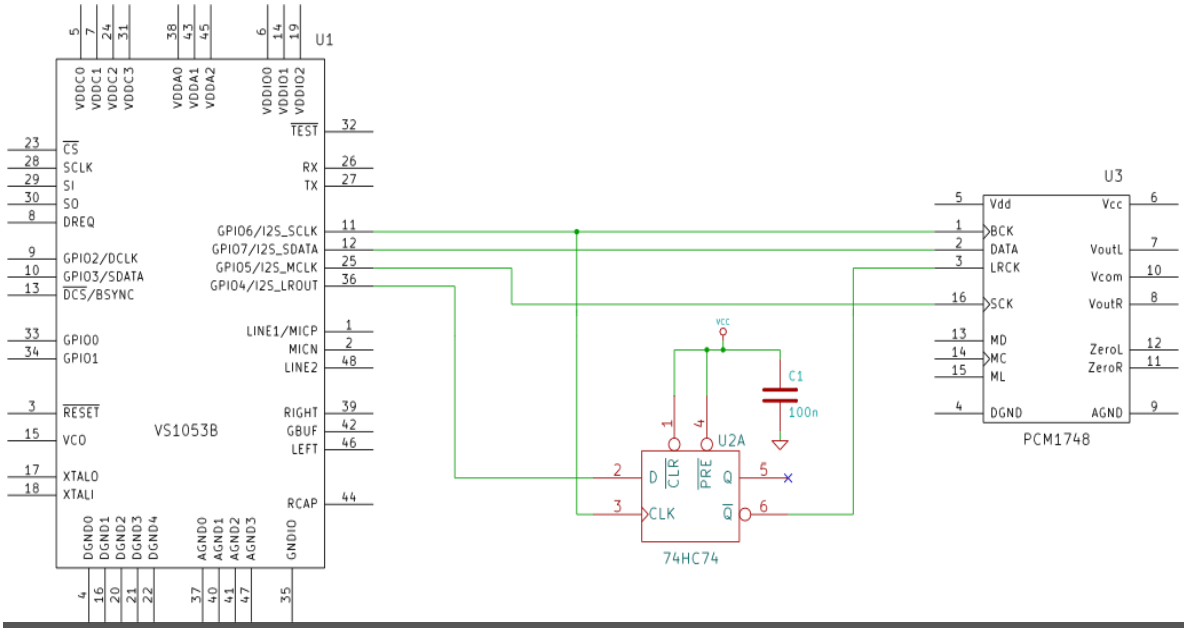
The VS1053 and VS1063 from VLSI Solutions are flexible digital audio decoders, supporting MP3, Ogg Vorbis, uncompressed PCM, and optionally AAC, WMA and FLAC. The decoders have integrated DACs (Digital to Analogue Converters) and an amplifier. If you want to keep the data in the digital domain, the decoders also offer I^2S outputs.

The only digital output format supported by the VS1053 and VS1063 is $32f_s$ I^2S. In summary: each of the left and right channels contains 16 bits of audio data per sample, and since there are two channels, the Bit clock runs at 32 times the Left/Right clock. This is what $32f_s$ means. If the Left/Right clock runs at 48kHz (a typical value), the Bit clock runs at 1.536MHz (48kHz×32).

The problem is that by far most DACs and Sample-Rate Converters (SRCs) only support I^2S starting from $48f_s$. The digital audio data may still be 16-bit, but it must be padded with eight extra bits to make it a total of 24 bits per channel (or 48 bits for a single stereo sample). The only $32f_s$ format that is commonly supported is Right-Justified.

DACs that have been confirmed to support $32f_s$ I^2S and that can therefore directly be used on the I^2S outputs of the VS10x3 decoders, are the Wolfson WM8741, the Cirrus Logic CS4398 and the Burr-Brown (Texas Instruments) PCM1780 series.

Fortunately, it is easy to convert $32f_s$ I^2S to ($32f_s$) Right-Justified. All that is needed is to invert the Left/Right clock and delay it by one Bit-clock cycle. Both can be done with a single D-type flip-flop.



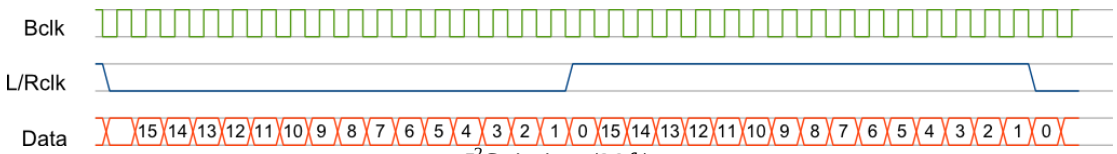
Converting $32f_s$ I^2S to $32f_s$ Right-Justified

This is a partial circuit, showing only how to connect the I^2S outputs of a VS1053 to the audio data inputs of a PCM1748. The PCM1748 is a high-quality DAC that supports $32f_s$Right-Justified —it also supports I^2S, but *only* $48f_s$ or $64f_s$.

We have also successfully used the above circuit to connect a VS1053 to SRC4190 and SRC4392 sample rate converters.

Walk-through

I^2S is a serial protocol with clock and data lines, plus a line to toggle between the left and right channels. There may also be a "master clock", but it is irrelevant for the discussion here. The Left/Right clock typically toggles between left and right at 48kHz. It is low for the left channel and high for the right channel.

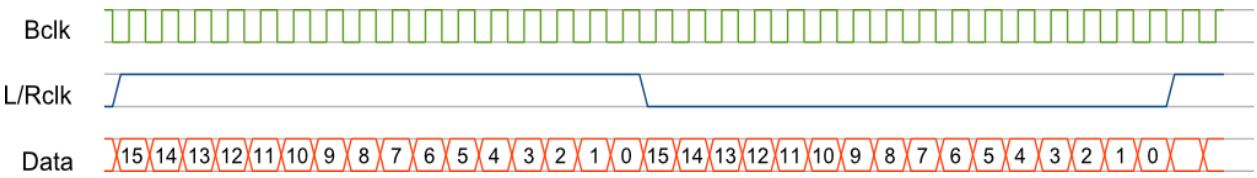


I^2S timing ($32f_s$)

The status of the Left/Right clock and the data are sampled at the rising edge of the Bit clock. The data is transmitted with the most significant bit first. All values are in two's complement. In particular, notice how there is one clock cycle between the toggling of the Left/Right clock and the first data bit for that channel.

Left-Justified and Right-Justified are very similar: there are the same Bit clock, data and Left/Right clock signals, although the status of the Left/Right clock is inverted relative to I^2S. The data is still transmitted with the most significant bit first, and it is also in two's complement.

However, the first bit of each channel is aligned with the toggling of the Left/Right clock —there is no offset of one clock cycle.



Right-Justified timing ($32f_s$)

The trick in converting I^2S to Right-Justified is to shift all data bits forward by one cycle of the Bit clock (shifting forward in time means: to the left in the diagram), which amounts to the same thing as to shift the Left/Right clock one cycle backward. The example circuit uses a 74HC74 dual D-type flip-flop (of which only one half is used). The Left/Right clock of the VS1053 and VS1063 is clocked in on a rising edge of the Bit clock. The output Left/Right clock follows the input clock, but since the propagation delay in the flip-flop, the DAC or SRC that is connected to the output of the flip-flop samples the Left/Right clock only on the *next* rising edge of the Bit clock.

As said, the output Left/Right clock must also be inverted from its input, because of the differences in the audio data formats of I^2S versus Right-Justified. This is accomplished by simply using the inverted output of the flip-flop.

The difference between Left-Justified and Right-Justified is the side on which the data bits are padded, but 16-bit 32fs audio, there is no padding.

Therefore, at 32fs, there is no distinction between Left-Justified and Right-Justified. The only reason to make this observation is that support for 32fs Left-Justified is very rarely mentioned in datasheets.

Note that D-type flip-flops are also available in small 6-pin or 8-pin packages, such as the NC7SZ74K8X from Fairchild's "Tiny Logic" series (or the equivalent NL17SZ74USG from On Semiconductor).

Concluding remarks

When working with digital audio, the goal is to stay in the digital domain as long as possible. When mixing multiple digital audio channels, by preference, this is done in the digital domain, rather than the analogue domain. When using AES/EBU or S/PDIF outputs, no conversion to the analogue domain is needed at all.

By supporting only $32f_s$ I^2S, the VS1053 and VS1063 digital audio decoders are limited in their interface to (external) DACs or SRCs. With a simple circuit, you have a wider choice of DACs and SRCs, because $32f_s$ Right-Justified is more generally supported by DACs and SRCs than $32f_s$ I^2S.

Source: http://www.compuphase.com/mp3/vs10x3_i2s.htm