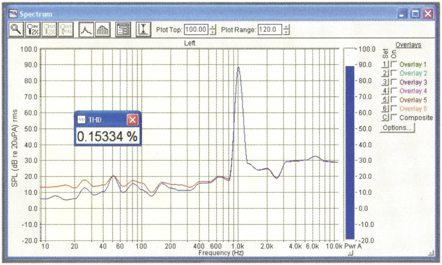
Mailbag: continued



Improving the Digital Audio Oscillator

While I agree with the magazine's response to the question "Improving the Digital Audio Oscillator" in the Ask SILICON CHIP pages of the February 2010 issue, I don't think the reply is as complete as it could be.

The Digital Audio Oscillator (SILICON CHIP, June 2009) uses a classic 8-bit passive 2R/R DAC followed by the op amp in question. In theory, if the waveform is accurately defined in the processor's lookup table, the maximum error should be ±1/2 bit, ie, a maximum quantisation error of 1 in 512.

By my calculations (ie, 20 x log(1/512)), that should give a "best possible" signal-to-noise ratio of -54dB or a best possible sinewave distortion of 0.2%. It turns out that this figure is achievable. The writer to Ask SILICON CHIP might have better put the question "Why is the oscillator's distortion no better than 3-4%? Will it help if I change the op amp?"

Your answer is correct – the LM358 is OK for this design but it is limiting the overall performance. The LM358's noise and distortion performance doesn't make things worse but its slew rate is a problem for the square-wave waveform at the top end. It can be as bad as 0.25V/µs rise/fall time according to the NatSemi specification sheet I looked at.

I have just finished designing and building a variation of this type of DDS (Direct Digital Synthesis) oscillator, using a cheaper ATtiny2313 micro and the same D/A ladder, followed by a TL072 op amp (with better slew rate!) with isolated main and inverted outputs. It also employs a way to get a reversed sawtooth/ramp waveform without the need to use another 256 bits of ROM. This gave much better distortion performance – 0.2%, as expected.

The attached spectrum sweep screen grab was taken using soundcard-measured SpectraPLUS demo software, so the measurement is hardly precise but the figures are likely to be within the ballpark. The 0.15% figure reported is probably better than expected because of the sharp filters in the sound card cutting off above 11kHz.

Inspection of the waveforms in the photos in the original SILICON CHIP article (June 2009, page 72) suggests the distortion problem probably lies in the waveform coding tables. The sinewave looks decidedly "peaky" and probably has a good load of second harmonic going on in there. The triangle wave clearly looks quite distorted too, as does the sawtooth. And if you can see it, the distortion is probably already hitting several percent.

The Altronics design could be considerably improved by two simple changes: (1) correct coding of the sinewave table in the ROM and (2) changing the LM358 to the TL072 or similar. In addition, unless the core code routine in the Altronics code is otherwise limited in some way, the oscillator should then also quite happily go up to 65kHz rather than the present design limit of 30kHz.

A further useful change might also be to swap the 78L05 regulator for an LP2950 which would give a modest improvement in battery life.

I made one other major addition in my oscillator. I hate pushbuttons for frequency selection (pressing UP/DOWN to select 27,345Hz for example, which you also can't do with this design, limited as it is to 500Hz steps above 1kHz), so I added a rotary encoder. It's got an SPST switch built-in, so pressing it down selects 1, 10 or 100Hz tuning steps anywhere across the 1Hz to 65,535Hz range, so tuning to 27,345Hz is much easier.

Finally, don't take this as criticism of the original article. It's a useful design and it spurred me on to crank out my version. I appreciate Altronics devoting the time to produce the kit. Really, this email is only intended to suggest that the design can probably benefit significantly from a couple of very low-cost changes.

Andrew Woodfield,
Christchurch, NZ.

Comment: good analysis, Andrew. We have passed your comments to Altronics.