◀1

Design Tools

Choose Your Averaging Method With Care When Acquiring Multiple Analog Channels

Tue, 10/10/2017 - 3:26pm   by Kendall Castor-Perry, Cypress Semiconductor

The proliferation of sensors and sources in electronic systems is driving equipment designers to acquire ever greater numbers of analog signal channels into their system MCU or sensor-fusion coprocessor. This is particularly true in the burgeoning small form-factor IoT space.

Signal averaging is a commonly-used technique in such data acquisition systems. It can enhance the usable resolution of numerical results and suppress several forms of noise. While a simple form of filtering, its overall effectiveness depends on the way averaging is implemented. This article contrasts two averaging techniques, one conventional (sequential averaging) and one more recently introduced (interleaved averaging).

DEEPER INSIGHTS

Critical Techniques for High-Speed A/D Converters in Real-Time Systems

WIRELESS DESIGN

Many contemporary mixed-signal MCUs and systems-on-chip incorporate averaging directly into their analog-to-digital converters' hardware. This can significantly reduce the amount of processing required by the MCU. This in turn simplifies coding and reduces how long the processor must operate in a high-power mode.

Whilst analog input multiplexers allowing the connection of several signals to the device are quite common, the hardware

averaging function in the majority of mixed-signal MCUs is carried out only on one signal channel at a time. When the averaging process is finished, an interrupt typically causes the firmware to select another analog input for conversion. In some devices, for instance the PSoC 4 family of programmable systems-on-chip processors from Cypress Semiconductor that has a 1 Msps 12-bit ADC, channel sequencing is built into the converter hardware. This allows a complete sweep of averaged channels to be completed with no processor intervention at all.

This well-established model of averaging multiple conversions of a single channel's signal before moving on to the next channel is called *sequential averaging*. The method has some limitations, chiefly to do with the slowing down of the available sample rate in a multichannel environment, not only for averaged channels but also for channels in the sequence that do not require averaging.

Recently, a new approach has become available that strengthens the arsenal of tools available to data acquisition system designers. The technique is called *interleaved averaging*, and it offers benefits both in systems where high frequencies are present in the signals to be sampled and when fast-sampled non-averaged channels are needed.

The distinction between sequential averaging and interleaved averaging is readily explained with some diagrams, in this case representing an eight-channel configuration. The raw ADC sample rate is set to 800 ksps, and sixteen of the 12-bit samples for each channel will be averaged together. This will result in a 16-bit output word, though the SNR contribution of the 12-bit sample quantization will limit the SNR to the equivalent of a 14-bit converter (on the assumption that the quantization noise contributions from each individual sample are uncorrelated).

Because there are eight channels, and each channel is sampled sixteen times to get a final result, the ADC needs to carry out 128 conversions to produce each set of results. This process takes 160 us, and result sets are available at a rate of 6250 per second.

The proliferation of sensors and sources in electronic systems is driving equipment designers to acquire ever greater numbers of analog signal channels into their system MCU or sensor-fusion coprocessor. This is particularly true in the burgeoning small form-factor IoT space.

Signal averaging is a commonly-used technique in such data acquisition systems. It can enhance the usable resolution of

numerical results and suppress several forms of noise. While a simple form of filtering, its overall effectiveness depends on the way averaging is implemented. This article contrasts two averaging techniques, one conventional (sequential averaging) and one more recently introduced (interleaved averaging).

Many contemporary mixed-signal MCUs and systems-on-chip incorporate averaging directly into their analog-to-digital converters' hardware. This can significantly reduce the amount of processing required by the MCU. This in turn simplifies coding and reduces how long the processor must operate in a high-power mode.

Whilst analog input multiplexers allowing the connection of several signals to the device are quite common, the hardware averaging function in the majority of mixed-signal MCUs is carried out only on one signal channel at a time. When the averaging process is finished, an interrupt typically causes the firmware to select another analog input for conversion. In some devices, for instance the PSoC 4 family of programmable systems-on-chip processors from Cypress Semiconductor that has a 1 Msps 12-bit ADC, channel sequencing is built into the converter hardware. This allows a complete sweep of averaged channels to be completed with no processor intervention at all.

This well-established model of averaging multiple conversions of a single channel's signal before moving on to the next channel is called *sequential averaging*. The method has some limitations, chiefly to do with the slowing down of the available sample rate in a multichannel environment, not only for averaged channels but also for channels in the sequence that do not require averaging.

Recently, a new approach has become available that strengthens the arsenal of tools available to data acquisition system designers. The technique is called *interleaved averaging*, and it offers benefits both in systems where high frequencies are present in the signals to be sampled and when fast-sampled non-averaged channels are needed.

The distinction between sequential averaging and interleaved averaging is readily explained with some diagrams, in this case representing an eight-channel configuration. The raw ADC sample rate is set to 800 ksps, and sixteen of the 12-bit samples for each channel will be averaged together. This will result in a 16-bit output word, though the SNR contribution of the 12-bit sample quantization will limit the SNR to the equivalent of a 14-bit

converter (on the assumption that the quantization noise contributions from each individual sample are uncorrelated).

Because there are eight channels, and each channel is sampled sixteen times to get a final result, the ADC needs to carry out 128 conversions to produce each set of results. This process takes 160 us, and result sets are available at a rate of 6250 per second.

This example also assumes that each channel has its own result register, as is the case for the PSoC 4 being used in this example. This is not the case for some mixed-signal MCUs, which have only one result register that must therefore be read out between channel changes.

Shown in figure 1, the behaviour of the standard sequential averaging solution in pseudocode form is as follows:

- flush the accumulation registers
- select channel 1
- take 16 samples at 1.25 us intervals, accumulate them in channel 1's register, total time 20 us
- select channel 2
- take 16 samples at 1.25 us intervals, accumulate them in channel 2's register, total time 20us
- and so on until all eight channels are done
- transfer the eight results with interrupt or DMA
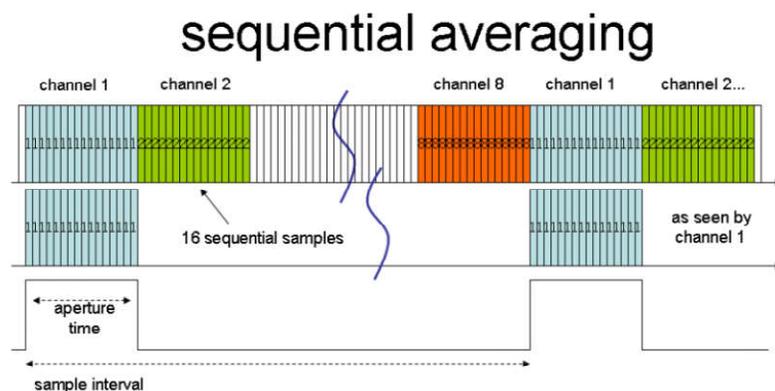
Figure 1: Sequential averaging

Each input is sampled in bursts of 16 conversions. The bursts for a given channel are 160 us apart, so effectively each channel is being sampled at 6250 samples per second. The sampling aperture – the length of time over which a channel is being sampled – is 20 us. This aperture introduces a lowpass filtering effect, but the bandwidth is quite high, with nulls in the frequency

response at multiples of (1/20 us), i.e. 50 kHz. This filtering is of no significant use in preventing aliasing. Any frequency components in the input signal close to multiples of 6250 Hz will be aliased down to near DC, possibly creating significant measurement noise. That could only be mitigated by pre-filtering each channel with its own analog antialiasing filter.

Also, there's a time offset between each channel, equal to 20 us. If cross-channel math functions need to be calculated (for instance, correlations or power calculations), this time difference will result in significant errors.

If you wanted to have unaveraged channels in such a sequence, the rate at which those channels can be sampled is dominated by the need to sequentially execute the averages of the other channels. So, despite having an ADC than can sample at 800 ksps, your unaveraged channel can only be sampled far less frequently. Ideally, you'd like to share that 800 ksps rate over your eight channels, get 100 ksps per channel, *and* keep the averaging.

Interleaved averaging – the solution to this conundrum – works differently (see Figure 2). The sequencer steps around the input channels as before, this time taking just *one* sample of each channel. After the channels have been stepped through N times, the output of all the accumulation registers can be read out.
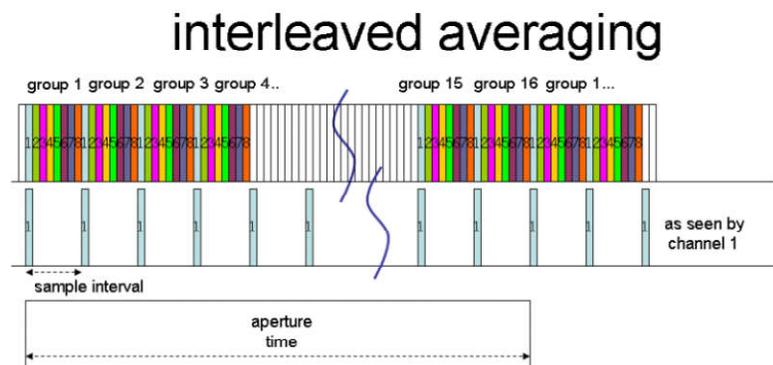

Figure 2: Interleaved averaging

The interleaved averaging sequence in pseudocode is as follows:

- set a hardware loopcounter to 0
- flush the accumulation registers
- repeat
- loopcounter += 1
- select channel [loopcounter]

- take a sample, accumulate it in channel [loopcounter]'s register, total time 1.25 us
- until loopcounter   = 8
- transfer the eight results with interrupt or DMA

The signal characteristics of this version of the averaging process are quite different. Now there are no bursts of conversions instead, each average is built from sixteen conversions spread uniformly over that 160 us once-round time and separated by 10 us. In other words, the per-channel sample rate is 100 ksps – the theoretical maximum when an 800 ksps ADC is shared across eight channels. The equivalent sampling aperture for this process is now the full 160 us, and this results in system frequency response nulls right at multiples of the final sample rate. The benefit of this is that no high frequency noise in the input signal can alias down to exactly DC. This makes for significantly more stable measurements, in turn greatly relaxing analog filtering requirements.

There is still a time offset between the channels, but now it has been reduced to 1.25 us. This is a far smaller fraction of the 160 us overall sample period, leading to far lower cross-channel calculation errors.

In this example, using interleaved averaging, this converter subsystem delivers samples of eight channels at an equivalent   14 bit SNR, at a 6.25 ksps rate, with good alias protection and very low interchannel time delay.

Interleaved averaging capabilities are being rolled out in new programmable system-on-chip devices from Cypress Semiconductor. Examples include the recently launched Cortex M4-based PSoC 6 and the PSoC Analog Coprocessor, which is an analog-focused member of Cypress's PSoC 4 family. The hardware in this device's ADC (fully configurable through the PSoC Creator Scan ADC component) also allows any channel in the sequence to be converted without averaging. In the example given earlier, this means that results from such channels continue to be available at a 100 ksps rate without affecting the timings of the averaged channels.

The ability to sample multiple channels at high resolution and high sample rate significantly expands the envelope of possibilities for cost-effective yet high performance analog signal capture using modern mixed-signal programmable systems-on-chip. When you need to convert multiple analog channels that may require averaging, bear in mind the signal processing effects of the

various averaging modes available in the devices you're considering.

About the Author

For nearly four decades, Kendall Castor-Perry has been chasing signals through electronic systems, wringing out the information they are hiding. He's a world-class authority on filters and precision analog circuit engineering and a tireless champion of the needs of the customer. He has been widely published and syndicated, especially when sharing his extensive filtering knowledge as "The Filter Wizard." He has a BA in Physics from Oxford and an MBA in MBA stuff from London Business School. Kendall is currently Senior MTS Architect in Cypress Semiconductor's Programmable Systems Division, pushing on the performance:power:price boundaries constraining tomorrow's critical sensor-processing systems.

RE E   NT PR D   T

RE TED RE DS
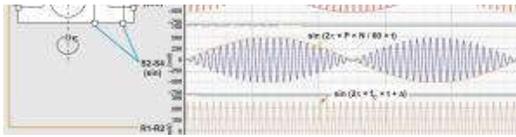
versampling ADCs ersus Comparators And Trac ing oop- ased Design Architectures or RDCs

ELECTRONICS
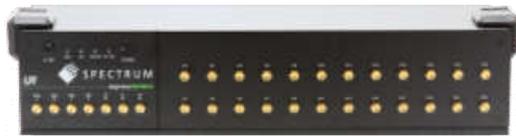
D Series f Digiti ers

WIRELESS DESIGN

ncreased Sensitivity ption Added To Digiti er roduct Series

WIRELESS DESIGN
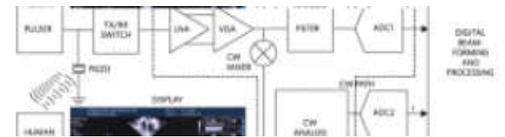
mprove ltrasound mage Clarity ith The Right ADC

MEDICAL

0 Comments   **Product Design & Development**   🔴 **Login** ▾

♡ **Recommend**   ↱ **Share**   Sort by Best ▾

Start the discussion…

LOG IN WITH   OR SIGN UP WITH DISQUS ⑦

Name

Be the first to comment.

ALSO ON **PRODUCT DESIGN & DEVELOPMENT**

Sensitivity to Time Improves Performance at Remotely Controlling Devices

1 comment • 9 days ago

Ranjith Obeyesekera — Most of the time, the problem related to fluctuation of voltage and time in receivers and transmitters.

Teleoperating Robots With Virtual Reality: Making it Easier For Factory Workers To

1 comment • 13 days ago

William K — The whole reason for "blue collar" workers is to be present with their human skills to do the work. They are more cost effective

Why Lab Researchers Should Talk With Industry Counterparts

2 comments • 9 days ago

wagnertinatlanta — The researchers had been trying to solve the wrong problem...So much of modern science is done not for practical

Microsoft Dives Deeper Into 'Mixed Reality' With VR Devices

3 comments • 14 days ago

William K — If we consider the present and past uses of other powerful narcotics we may decide to back away quickly.

✉ **Subscribe**   Ⓓ **Add Disqus to your site**Add DisqusAdd   🔒 **Privacy**