



BY BONNIE BAKER



## Analog versus digital: riding the fence

**A**s an analog-world descendant, I always hear comments in the hallway about how digital designers don't really understand analog issues. Digital designers will go so far as to unsympathetically say the same about analog-IC designers. There is no bridge between these two camps unless the participants ride the fence and enter the mixed-signal domain together.

True to the analog spirit, not all data converters use the same digital format. Some converters use unsigned-binary-data types; other converters use two's-complement signed data. To even further complicate matters, some converters produce 12- or 14-bit output words, and others produce 16-bit output words. Yet another technology is the 24-bit delta-sigma converter.

Forget the reasons for these analog-design decisions. With all of these converters, the location of the ADC LSB is in the processor's 0-bit location within the 8-, 16-, or 32-bit word. This situation makes perfect sense to an analog designer. However, the signed-bit of a 12-bit converter resides in position 11 in the processor. If you assign a 16-bit-wide C variable to the converter's output word, C assumes that the sign bit is in position 15. The processor does not recognize a negative number from the converter and assumes that all codes from the 12-bit, bipolar-in ADC are positive. This situation occurs because the signed bit is in the wrong position.

You can approach this problem in several ways. The first, a read-modify-write approach, shifts data in the

processor register. The CPU reads the data, shifts the bits in the register containing the data by the necessary amount of bit positions, and writes the data back to memory. A DSP can complete this shift with one cycle. A controller requires many cycles for this shift process. If this strategy is the one you choose, be careful of cache incoherency. Cache incoherency means that the cache is unaware that the DMA controller has placed new words in memory. As a result, the CPU instead shifts the old contents of the cache. It is important to remember that whenever you implement a bit shift to the left, you multiply the ADC results by two.

Another option is to shift the data into the right position in a processor loop. On the downside, this approach uses the CPU and requires additional MIPS. Another alternative is to directly connect the converter to the processor's data bus. If you connect bit 11 of the 12-bit converter to bit 15 of the processor's data bus, the sign bit will then be in the right place. Complete the data-acquisition task by zeroing register bits 11 through 14. Otherwise, these bits are indeterminate. This approach is pos-

sible only with parallel interfaces. In this scenario, the DMA-based transfer need not shift data later on.

Serial-port users are more fortunate than parallel-port users because most of the serial ports in processors offer the receiving feature of "left justify and zero fill LSBs" or "right justify and sign-extend MSBs." This feature significantly reduces the amount of work the CPU must perform once the data is in memory, with one drawback. Some serial ports work only if the received word length is a power of two, such as a 16-bit word length. This receiving feature does not work with 12-, 14-, or 24-bit values.

Analog designers can help with the processor-interface problem. All the processors I know about have 8-, 16- or 32-bit data types, but I have never heard of a 12- or 14-bit data type. Until all of the analog-chip designers take the leap and straddle the fence, don't take things for granted when you are designing the digital interface to a converter. Read the ADC data sheet and verify the bit positions in the transmission across the digital interface. If you do this initial examination of the ADC's documented digital interface, you will be more successful with your first spin on the design. **EDN**

*Special thanks to my digital buddy, Richard Oed, systems engineer with the data-acquisition-products group at Texas Instruments.*

### REFERENCES

- 1 Oed, Richard, "Let's Talk! Ten things to remember if your data converter is to understand what a DSP is saying to it," *New Electronics*, March 2005, <http://direct.bl.uk/bld/PlaceOrder.do?UIN=164814814&ETOC=RN&from=searchengine>.

*Bonnie Baker is a senior applications engineer at Texas Instruments and author of A Baker's Dozen: Real Analog Solutions for Digital Designers. You can reach her at bonnie@ti.com.*