

Use an RC filter to 'deglitch' a DAC

EDN edn.com/use-an-rc-filter-to-deglitch-a-dac/

Bonnie Baker

July 28, 2011

Annoying speed bumps, which have become ubiquitous on roadways, force you to either slow down or find a way around them. While driving recently, I needed to negotiate one of those bumps, and doing so reminded me of a precision, 16-bit DAC using an R2R architecture, which places resistors in parallel as a resistor ladder (**Figure 1**). The DAC had a glitch at midscale. You can add deglitching circuits to DACs such as these ones that share traits with speed bumps: To avoid them, you must either slow down or find a way around them. Two common DAC-deglitching circuits accomplish these tasks. Simple lowpass filters represent a slowdown tactic, and sample/hold circuits offer a way to avoid the glitch. These deglitching circuits can either decrease the glitch's amplitude or remove its energy.

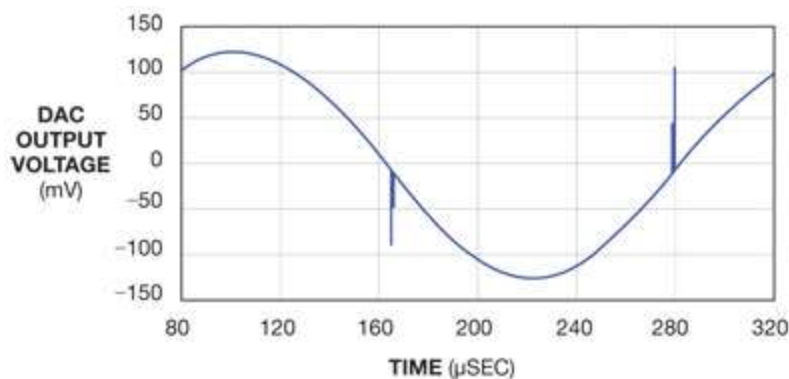


Figure 1 You can use an R2R architecture, which places resistors in parallel as a resistor ladder, to deglitch a DAC.

The simplest DAC-deglitching method uses an RC filter (**Figure 2a**) at the DAC's amplifier output, V_{OUT} . This filter attenuates the amplitude of the glitch and increases the settling time. In **Figure 2b**, the top curve (red) is the signal on the DAC's LDAC (load-DAC) pin. You serially load a data word into the DAC using the DIN (data-in) and clock (CLK) pins. Once the DAC has a complete word of data, the rising edge of the LDAC pin loads the data word into the internal DAC registers. This action initiates a change in the DAC's output voltage. The middle

curve (yellow) shows the measured midscale analog glitch from the DAC's output. The bottom curve (green) shows the measured analog signal after an RC lowpass filter.

When you increase or decrease the data-code value, the output voltage also increases or decreases. But at midscale, one-fourth scale, and three-fourth scale, the DAC generates a glitch—the midscale glitch being the largest. To determine the appropriate RC ratio, examine the glitch period and select a 3-dB point for your filter approximately one decade lower than the glitch frequency.

For example, the glitch period in **Figure 2b** is approximately 1 μsec . This value translates to a 1-MHz glitch period. From this estimate, the RC values in **Figure 2** create an 80-kHz lowpass filter. When selecting your RC values, make sure that the resistance is low enough to avoid loading errors. This RC filter solves the R2R DAC-glitch problem, but there is no free lunch. As you can see from the bottom curve in **Figure 2b**, the RC filter extends the DAC's settling-time output signal.

TALKBACK

We would love your feedback!
Post a comment below.

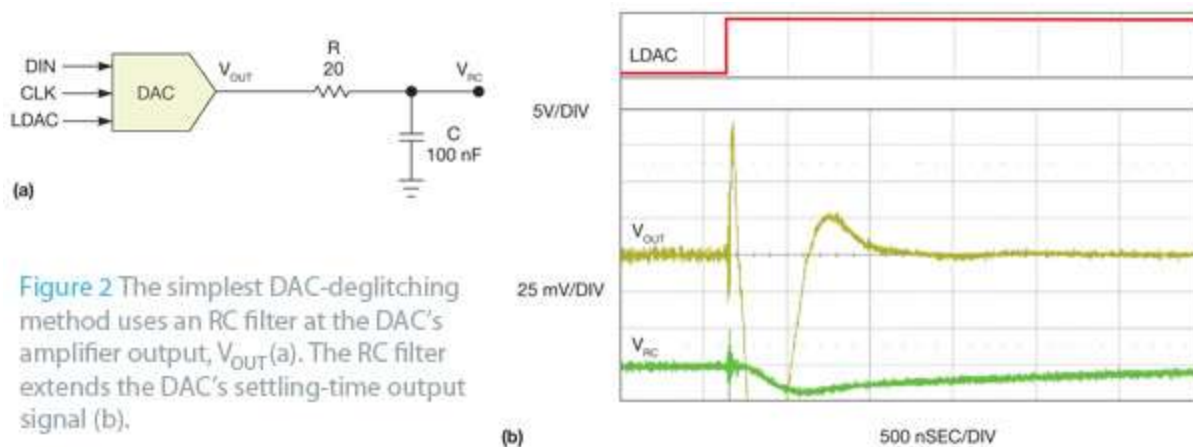


Figure 2 The simplest DAC-deglitching method uses an RC filter at the DAC's amplifier output, V_{out} (a). The RC filter extends the DAC's settling-time output signal (b).

Depending on your application's requirements, a simple RC filter may do the trick. If the system calls for an R2R DAC that has an interfering glitch, you might be able to combine a switching capacitor with an RC filter to solve the problem.

References

1. Baker, Bonnie, "What's a little glitch among friends?" *EDN*, Dec 15, 2006, pg 28.
2. Baker, Bonnie C, "A DAC for all precision occasions," Texas Instruments, *Analog Applications Journal*, 2008.

Acknowledgment

The author wishes to give special thanks to Tony Calabria, a precision-analog-applications engineer at Texas Instruments.

Bonnie Baker is a senior applications engineer at Texas Instruments.