Gray-code Digital-to-Analog Converters (DACs) for Glitch Reduction

Simon Voigt Nesboe, Advisor: Prof. Xingguo Xiong
Department of Electrical Engineering
University of Bridgeport, Bridgeport, CT

Abstract

Digital-to-analog converters (DACs) are widely used for signal processing and other applications. A popular Digital-to-Analog Converter (DAC) topology is the R-2R ladder. It is a simple topology which consists of only two resistors and a switch per bit, and typically one OPAMP buffer at the output. However, such a topology is vulnerable to glitches (voltage spikes) that may occur when several bits in the digital input change at the same time. Although a DAC is often followed by a Reconstruction Filter (RCF), this filter may need to be of a high order to attenuate the voltage spikes sufficiently. Such a filter increases the size of the die in an integrated circuit (IC), or requires more spaces, components and increases cost for a printed circuit board implementation. In some applications, such as an analog control signal to a valve in a control system, the filter may not even be necessary, but it is probably desirable to not have glitches.
One common way of deglitching is to use a Track-and-Hold Amplifier (THA) at the output, but that again requires extra circuitry to tell the THA when to hold, or circuitry to detect when the inputs are changing.
In this work some alternative topologies that eliminate glitches are considered, primarily DACs based on R-2R ladders that accept Gray code input.

DAC with Glitch Reduction

Glitches in data converters are typically caused by two phenomena [3]:
1) Capacitive coupling
2) Differences in how fast the switches open and close
Typically the glitch behavior is dominated by the differences between the switches. Unlike capacitive coupling, which often leads to both a positive and negative spike, the glitch due to the switches is usually uni-polar, meaning that there is one voltage spike that is either positive or negative, not both. An example is shown in figure 1. The area of the glitch spikes is often used as an estimate of a DACs glitch performance, and is sometimes referred to as glitch energy, although the correct unit is volt-seconds. [3].
It appears as if there is no well defined method of calculating the glitch area, because the calculated area depends on several factors that are themselves not well defined such as settling time, which is when the output has settled to within an arbitrary precision, e.g. 90%. How the area is calculated is another matter, many manufacturers use net glitch area, where positive and negative pulses are added and cancel each other out. If the output resembles an under-damped response, with positive and negative, the net glitch area is approximated by considering the first 4 pulses only (as seen in figure 1a), and can be further approximated by considering the pulses to be triangles. If the glitch consists of positive and negative pulses of equal weight, then the net glitch area may become very small, so obviously this may be an optimistic way of measuring glitch performance. A different approach is to specify the glitch performance by measuring the peak glitch area, which is the area of the biggest glitch pulse.

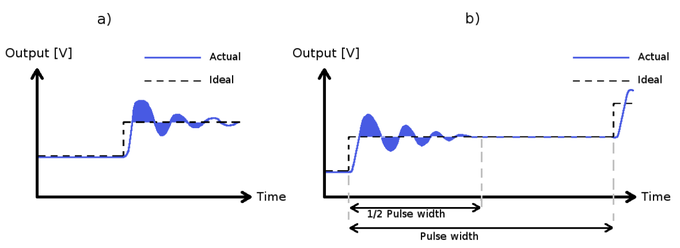

*Figure 1: **a)** Example DAC output with a code change. The four shaded areas are used to calculate the "net glitch area". **b)** Example of how the software uses all over/undershoot before 1/2 of the pulse width to calculate glitch areas.*

As a part of this thesis a software was developed that calculates, among other things, glitch performance of a data converter. This software will run SPICE simulations of a DAC, where each code is tested from the lowest to highest. The user should select a pulse width much higher than the settling time. The final output value for a code is then taken by looking at the output after one half of the pulse width. The software then goes back to the beginning of this particular code, and looks for the first time, if at all, that the output value goes beyond the final value. The glitch areas are calculated from that point up till half the pulse width, by summing the output value minus the final output value, for each time step. The sum is then multiplied by the simulation time step, so the area calculation is essentially done by means of a Riemann integral.

Discussion

One of the most common Digital to Analog Converter (DAC) topologies is the R-2R ladder (see fig. 2 below). For each bit, the R-2R ladder consists of a two-pole switch and two resistors, one of size R, and the other of size 2R. The chain is terminated by an additional 2R resistor, and an OPAMP buffer is typically used. The obvious benefit of this topology is that the size of the circuit scales linearly with the amount of bits. While the bit width can be increased by adding an additional R-2R, the matching of the resistors typically limit the accuracy that can be achieved with the R-2R ladder. Glitches, in the form of voltage spikes, can occur in the R-2R ladder when several switches change state at the same time. As explained in [2], the predominant case is when the Most Significant Bit (MSB) has a different value than all the other bits, and the input code changes such that all bits change value. E.g. for an 8-bit DAC, going from 10000000 to 01111111, or 01111111 to 10000000.

Two interesting topologies were presented in [1], which are R-2R ladders that have been altered in a way such that they can accept Gray Code input (fig. 3). By using Gray Code the glitches described for the R-2R ladder can be reduced, for a signal that increments with only 1 value at a time, because only 1 bit changes at a time in Gray Code for increments of 1. Little research is available on the topic of Gray-code DAC ladders. The patent in [1] dates back to 1986, and has expired by now, so the topologies described in patent are free to be used. There are some academic papers that describe DAC architectures that utilize Gray-code in some way, but they are rather application specific, and none of them have the simplicity of the R-2R ladder. Because of the little attention that has been devoted to this topic, this work seeks to explore the gray code R-2R ladder and similar topologies. The goal is to:
1) Discover how efficient it is for decreasing glitches
2) Declare if it is a viable option for decreasing glitches, compared other DAC topologies and deglitching techniques
3) Develop it further into a gray code I-2I current steering DAC.

Results

The first item in the list above has already been proved to some degree. Using the circuit simulator LTSpice, a circuit for a traditional R-2R DAC and one for the gray code R-2R DAC were designed and simulated. The opamps used were ideal opamps but with selectable open loop gain and bandwidth, set to 100k and 100MHz, respectively. The switches used were not ideal, but based on a SPICE model with selectable on/off resistances and rise/fall times. Of course in a real design the switches and opamps has to be designed and implemented with transistors, but that is still a work in progress.
The simulations were run with several different parameters, but the results were always as seen in figure 4; large glitches were observed for the traditional R-2R ladder, but the gray code DAC had little or no glitches. Clearly this topology is effective in reducing glitches, the next step now is to see if a practical design is possible.

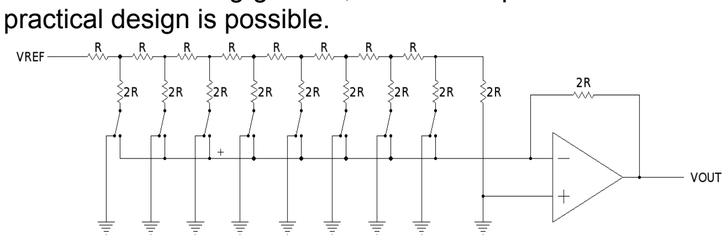

Figure 2: A traditional R-2R DAC.

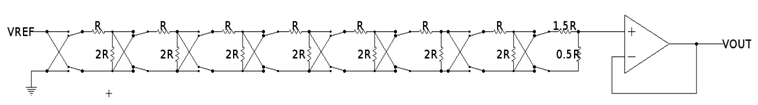

Figure 3: A gray code R-2R DAC, as described in [1]. Each input bit controls a pair of two-pole switches, and the input has to be gray coded.

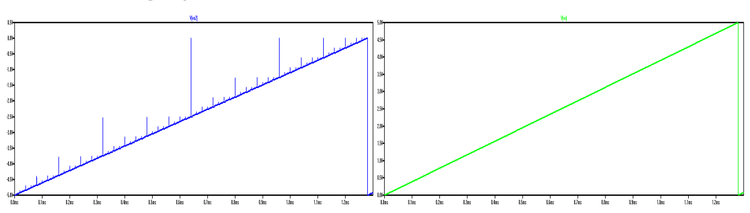

Figure 4: Transient simulation of the R-2R DAC (left), and gray code R-2R DAC (right), with input tamping up from 0 to 255. One can clearly see the glitches in the regular R-2R DAC, where as in the gray-code DAC they appear to be absent.

Conclusion and future work

The simulation results showed that the gray code R-2R DAC greatly reduces glitches. The results were obtained using SPICE models for the Opamp and switches though, so before any final conclusions can be drawn a full transistor level circuit should be design, and then a real circuit should be fabricated and tested.

Another important point that will be looked into is the conversion to gray code. Since practically all digital systems use natural binary numbers, at some point this conversion is needed. Care must be taken when designing a gray to binary converter, because if there are glitches or intermediate states in the digital gray output, then more than one switch will be active at a time in the ladder and glitches are likely to appear, rendering the whole gray code ladder useless.

A new circuit based on the gray code R-2R DAC in figure 3 has been devised. The basic idea is to remove the R-resistors and replace the 2R resistors in fig. 3 with binary-weighted current sources. The result is shown in figure 5. This circuit, using ideal current sources, was simulated in LTSpice and the results were similar to those of the gray code R-2R DAC in fig. 4 (right). Current steering DACs have some advantages over resistor DACs, namely they do not need buffers (although it is drawn in fig. 5) to drive resistive loads, and they are suitable for high frequency applications. [2] Glitches are a particular problem for current steering DACs without buffers, since the current is being routed directly to the output [2] it is not possible to use alternative deglitching techniques such as the THA mentioned in the abstract. Therefore this is a very interesting result that could make high-speed glitch-free current-steering DACs possible.

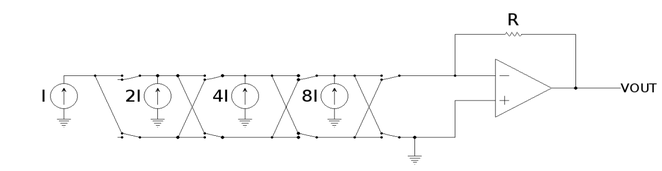

Figure 5: Current-steering gray-code DAC

Work is currently being done on a VLSI implementation of the gray code DAC. A folded cascode opamp has been designed, fig. 5, with a constant-transconductance biasing circuit. The opamp has wide swing common mode input range, which was deemed important since a DAC must produce outputs at both extremes of the voltage range. Parts of the layout so far is seen in fig. 6.
The goal is to have a chip manufactured with the different DAC topologies discussed, so that their real world performance can be compared.

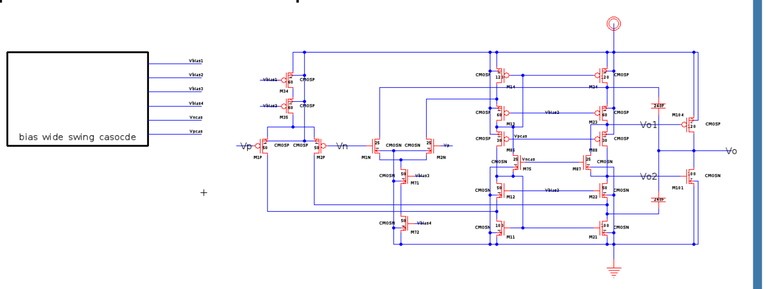

Figure 6: Folded cascode opamp schematic, with wide-swing cascodes, and PMOS and NMOS input differential pair for wide common mode input range.

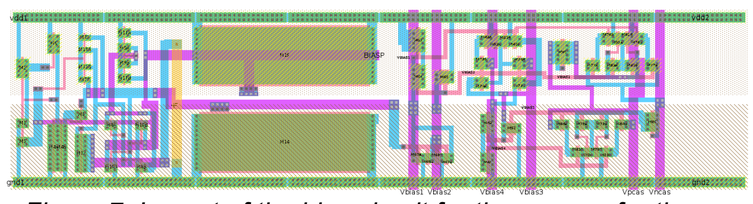

Figure 7: Layout of the bias circuit for the opamp, for the C5N process.

References

[1] N.C. Seiler. *Gray code DAC ladder*, US Patent 4,591,826, 1986.

[2] R. J. Baker. *CMOS: Circuit Design, Layout, and Simulation,* 3rd edition, Wiley-IEEE press, 2010. ISBN 978-0-470-88132-3

[3] W.A. Kester and inc Analog Devices. *Data Conversion Handbook*. Analog Devices series. Elsevier, 2005.