

Full digital driving voice and audio load on an IC

Dake Liu¹ Ronny Nilsson and Fredrik Norling²

¹Dept. of Electrical Engineering, Linköping University, SE-581 83 Linköping, Sweden
dake@isy.liu.se, phone +46 13 281256, fax 46 13 139282

²Ericsson Microelectronics, Isafjordsgatan 16, 16481, Kista-Stockholm, Sweden

Abstract -- This paper presents a study of how digital to analog conversion for low cost voice-audio can be achieved without analog components. The idea is to use an 1-bit oversampling digital to analog converter and utilize the fact that a loudspeaker has a very limited frequency bandwidth. Several methods to implement 1-bit digital to analog converters are thoroughly discussed. Functional design and verification as well as low power design are done. The design is implemented in a 0.35 μ full-custom ASIC. The measured power consumption is 0.056mW when supply voltage is 2.5V. The solution is suitable for voice and audio terminals for communications.

I. INTRODUCTION

Design cost and silicon cost of an digital to analog converter is high. The reason is that conventional D/A converters need precise analog components in their conversion circuits and in their analog filters. For example, if a 13-bit D/A converter with a supply voltage of 2.5V is to be designed, the error (1/2LSB) tolerance on voltage must be less than 0.15mV. We need a special analog silicon technology to support such high requirements and an analog process cost could be 25% higher comparing to the digital process with the same feature size.

A low power driving on IC (Integrated Circuit) for voice and audio load is essential components in a communication terminals IC or in a home electronic product IC. After digital signal processing, we use to drive the voice and audio load using analog power amplifier via a DA (Digital to analog) converter. An ideal case is that, a low power voice and audio load or an earphone should be driven only by digital circuit based on digital silicon process to decrease the cost from DA converter and the cost from the analog power amplifier. The design and silicon cost should be very low and the performance of driving voice and audio load must be high.

By a proper design, Δ/Σ modulators can be used to modulate the pulse width representing the energy of voice-audio signal and the modulator generated noise on higher

frequency cannot be realized by human as the noise is up to the ultra sonic range. Design based on the concept above will induce a full digital driving circuit without any off chip component associate to the voice and audio load in our solution in this paper. We use only a number of driving pads on chip adapt to the impedance and the power of the voice and audio load.

Δ/Σ modulation theory has been well developed. In this research, we reviewed, analyzed, and selected algorithms to get low power consumption, high stability, and low silicon area. We focused on the implementation and got a satisfied result.

II. ALGORITHMS

Whenever a signal resolution is decreased quantization noise arise. This happens when an analog signal is A/D converted or when the number of bits representing a digital signal is reduced. If the input signal $X(n)$ is assumed to be varying rapidly in time then the quantization error $e(n)$ can be approximated as an independent signal, which is varying between $\pm\Delta/2$. Δ is the smallest quantization step, i.e. $LSB(2^{-N})$. With the assumption that $e(n)$ is randomly, its probability function will be constant. Using stochastic theory the quantization noise power $P_e = e_{rms}^2(n)$ can be calculated:

$$P_e = \int_{-\frac{f_s}{2}}^{\frac{f_s}{2}} S_e(f) df = \int_{-\frac{f_s}{2}}^{\frac{f_s}{2}} k_x^2 df = k_x^2 f_s = \frac{\Delta^2}{12} \quad (1)$$

Which yields the height k_x of the quantization spectrum:

$$k_x = \frac{\Delta}{\sqrt{12}} \sqrt{\frac{1}{f_s}} \quad (2)$$

Here, f_s is the oversampling frequency. There are two ways to decrease the height of quantization noise density $S_e(f)$:

1. Increase the number of quantization levels. The noise power decreases 6dB for each bit, that is Δ is halved for each bit.

- Increase the sampling frequency f_s . The height k_x will decrease with higher frequency, although the total noise power is not decreasing.

As shown the height of the quantization noise spectral density can be decreased when the sampling frequency f_s is increased. This lead up to examine how the signal to noise ratio (SNR) is affected by increasing over-sampling frequency. The SNR is defined as

$$SNR_{max} = 10\log\left(\frac{P_{signal}}{P_{noise}}\right) = 20\log\left(\frac{x_{rms}}{e_{rms}}\right) \text{ dB} \quad (3)$$

Assuming that the input signal $x(n)$ is a random signal uniformly distributed between 0 and 1. Then it can be shown (in similarity to e_{rms}) that

$$x_{rms} = \frac{1}{\sqrt{12}} \quad (4)$$

So that

$$SNR = 20\log\left(\frac{\frac{1}{\sqrt{12}}}{\frac{\Delta}{\sqrt{12}}}\right) = 20\log\left(\frac{1}{\Delta}\right) = 20\log(2^N) = 6.02N \text{ dB} \quad (5)$$

It implies that a 13-bit quantization has a SNR of 78 dB.

A. Oversampling and noise shaping

A common solution to reduce errors in systems is to use some sort of error feedback. The system in the following figure shows the basic principle.

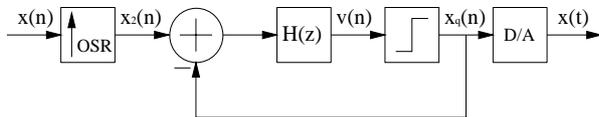


Figure 1

As in the figure, the input signal is interpolated. Instead of quantizing the interpolated signal, it is processed by $H(z)$. The D/A is an analog filter to smoothen the pulse from the 1-bit quantizer. The quantizer and the system $H(z)$ is denoted as noise shaping loop, NSL. The NSL and its linear model is shown in figure 2.

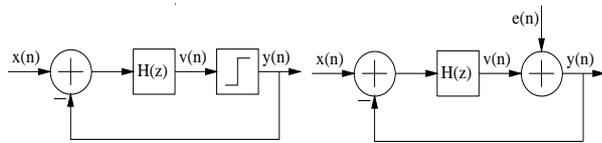


Figure 2

Analyzing the linear model, the transfer functions for the signal $x(n)$ and the quantization error $e(n)$ can be derived

$$\left. \begin{aligned} Y(z) &= V(z) + E(z) \\ V(z) &= (X(z) - Y(z)) \cdot H(z) \end{aligned} \right\} \quad (6)$$

$$Y(z) = \frac{H(z)}{1 + H(z)}X(z) + \frac{1}{1 + H(z)}E(z)$$

By denoting

$$S_{TF}(z) = \frac{H(z)}{1 + H(z)} \quad \text{Signal Transfer Function}$$

$$N_{TF}(z) = \frac{1}{1 + H(z)} \quad \text{Noise Transfer Function}$$

The above equation can be written as

$$Y(z) = S_{TF}(z)X(z) + N_{TF}(z)E(z) \quad (7)$$

It is preferred to choose $H(z)$ so that the signal is unaffected, i.e. $S_{TF}(z)$ should be a all-pass filter or low-pass filter. In order to move the quantization error out of the signal frequency bandwidth, $H(z)$ should be chosen so that $N_{TF}(z)$ acts as a high-pass filter. Notice that $N_{TF}(z)$ has its zeros where $H(z)$ has its poles. So by letting $H(z)$ have high magnitude in the signal frequency bandwidth, the signal will be left unaffected but the quantization noise will be attenuated.

B. The first order noise shaping

In order to attenuate the quantization noise in the signal bandwidth the $N_{TF}(z)$ needs a zero at DC, i.e. $z=1$. This implies that $H(z)$ needs a pole at DC

$$H(z) = \frac{1}{z-1} \quad (8)$$

therefore

$$S_{TF}(z) = \frac{H(z)}{1 + H(z)} = \frac{\frac{1}{z-1}}{1 + \frac{1}{z-1}} = z^{-1} \quad (9)$$

$$N_{TF}(z) = \frac{1}{1 + H(z)} = \frac{1}{1 + \frac{1}{z-1}} = 1 - z^{-1} \quad (10)$$

The $S_{TF}(z)$ is simply a time delay and $N_{TF}(z)$ will function as a high-pass filter. Substituting $z = e^{j\omega T}$ gives

$$N_{TF}(\omega) = 1 - e^{-j\omega T} = \frac{e^{\frac{j\omega T}{2}} - e^{-\frac{j\omega T}{2}}}{2je^{\frac{j\omega T}{2}}} = \sin\left(\frac{\omega T}{2}\right) 2je^{\frac{j\omega T}{2}}$$

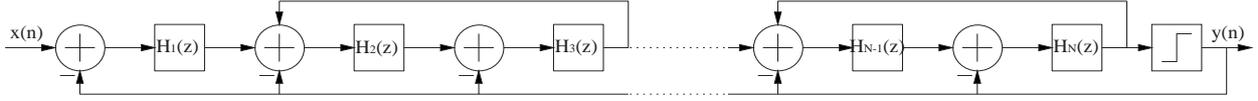


Figure 3

To examine the magnitude, the absolute value is calculated

$$|N_{TF}(\omega)| = 2 \sin\left(\frac{\omega T}{2}\right) \quad (11)$$

One can see that the noise is actually amplified at higher frequencies, but keep in mind that the original signal has been interpolated, so that the signal bandwidth is below 0.05 of the normalized angular frequency. At such low frequency the quantization noise is attenuated. This leads up to examine how much of the quantization power there is in the signal bandwidth (0 to f_x). Substituting $\omega T = \frac{2\pi f}{f_s}$ and yields

$$|N_{TF}(f)| = 2 \sin\frac{\pi f}{f_s}$$

Let $N_{TF}(z)$ act as a filter for the quantization noise to calculate the in-band noise

$$P_e = \int_{-f_x}^{f_x} S_e^2(f) |N_{TF}(f)|^2 df = \int_{-f_x}^{f_x} \left(\frac{\Delta^2}{12}\right) \frac{1}{f_s} \left[\sin\left(\frac{\pi f}{f_s}\right)\right]^2 df \quad (12)$$

Now assume that the oversampling rate is reasonably high ($f_s \gg 2f_x$), then $\sin\left(\frac{\pi f}{f_s}\right) \approx \left(\frac{\pi f}{f_s}\right)$

$$P_e \approx \left(\frac{\Delta^2}{12}\right) \left(\frac{\pi^2}{3}\right) \left(\frac{2f_x}{f_s}\right) = \frac{\Delta^2 \pi^2}{36} \left(\frac{1}{OSR}\right)^3 \quad (13)$$

If the signal, as before is assumed to be sinusoidal, the SNR is given by

$$\begin{aligned} SNR_{max} &= 10 \log \frac{P_s}{P_e} = 10 \log \frac{\frac{1}{8}}{\frac{\Delta^2 \pi^2}{36} \left(\frac{1}{OSR}\right)^3} = \log \left(\left(\frac{3}{2}\right)^{2N} \left(\frac{3}{\pi^2} (OSR)^3\right) \right) \\ &= 6.02N + 1.76 - 5.17 + 30 \log(OSR) \text{ dB} \end{aligned}$$

Trying to do the 13-bit to 1-bit conversion again will give

$$\begin{aligned} 80 \text{ dB} &= 6.02 + 1.76 - 5.17 + 30 \log(OSR) \\ OSR &= 400 \text{ times} \end{aligned}$$

Remember that signal bandwidth is $f_x=4\text{kHz}$ gives

$$f_s = OSR \cdot 2f_x \approx 3.2 \text{ MHz}$$

Through this is the frequency realizable in today hardware technologies, the power consumption based on this OSR is not be optimized. We therefore, looking for higher order noise shaping to decrease the OSR.

C. High order noise shaping

To get even lower OSR and still get a high SNR, we increased the order of the NSL. The following figure shows a N-th order NSL.

Notice the positive feedbacks that acts as resonators and places zeros in $N_{TF}(z)$. However the complexity in high-order ($N>2$) requires more hardware and therefore for this design the requirement of power consumption (less than 0.1 mW) is hard to fulfill. Also stability theory for NSLs with order $N \geq 3$ is very poorly developed and the designer is left to empirical methods. Therefore NSL with order $N \geq 3$ is not further examined in this paper.

D. Stability

In order to make the quantization valid the quantizer must be prevented from being overloaded. An overloaded quantizer is a quantizer which quantization error is greater than $\pm \Delta/2$, i.e. 1-bit quantizer is overloaded if the input magnitude is greater than 2. First-order NSL can be shown to be stable for all input signals with magnitudes less or equal to 1. Stability theory for high-order NSLs is not very well developed. However a rule of thumb exists [2]

$$|N_{TF}(\omega)| \leq 1.5 \quad 0 < \omega < \pi$$

This rule has a tendency to discard stable NSLs as it is quite rigorous. Since high-order NSLs are not analyzed in this paper no further analyze concerning high-order NSLs ($N \geq 3$) will be done.

E. Second order noise shaping loop

To meet the requirements from the specification the first-order NSL is discarded due to the SNR requirements and high-order NSLs due to their complexity and their stability problems. This left the second-order NSL as the only solution. The following figure shows a second-order NSL.

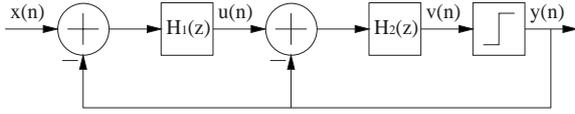


Figure 4

Replace the quantizer with a linear model and calculate the transfer functions

$$\left. \begin{aligned} Y(z) &= V(z) + E(z) \\ U(z) &= (X(z) - Y(z))H_1(z) \\ V(z) &= (U(z) - Y(z))H_2(z) \end{aligned} \right\}$$

$$Y(z) = \frac{H_1(z)H_2(z)}{1 + H_2(z)(H_1(z) + 1)}X(z) + \frac{1}{1 + H_2(z)(H_1(z) + 1)}E(z)$$

Again, $H_1(z)$ and $H_2(z)$ are chosen as

$$H_1(z) = \frac{z}{z-1} \quad (14)$$

$$H_2(z) = \frac{1}{z-1} \quad (15)$$

then $S_{TF}(z)$ and $N_{TF}(z)$ can be written as

$$S_{TF}(z) = \frac{H_1(z)H_2(z)}{1 + H_2(z)(H_1(z) + 1)} = z^{-1} \quad (16)$$

$$N_{TF}(z) = \frac{1}{1 + H_2(z)(H_1(z) + 1)} = (1 - z^{-1})^2 \quad (17)$$

The $S_{TF}(z)$ is the same as in the first-order NSL but the order of $N_{TF}(z)$ has increased by one. Using similar way presented before in this paper, we get the result of in-band noise

$$P_e \cong \frac{\Delta^2 \pi^4}{60} \left(\frac{1}{OSR} \right)^5 \quad (18)$$

And the maximum SNR when the 13-bit to 1-bit conversion is done resulting in

However this is a bit optimistic result since so far no stability tests has been made.

$$SNR_{max} = 10 \log \left(\frac{P_s}{P_e} \right) = 10 \log \left(\frac{\frac{1}{8}}{\frac{\Delta^2 \pi^4}{60} \left(\frac{1}{OSR} \right)^5} \right)$$

$$= 10 \log \left(\left(\frac{3}{2} \right)^{2N} \left(\frac{5}{4} (OSR)^5 \right) \right)$$

$$= 6.02N + 1.76 - 12.9 + 50 \log(OSR) \cong 51 \text{ dB}$$

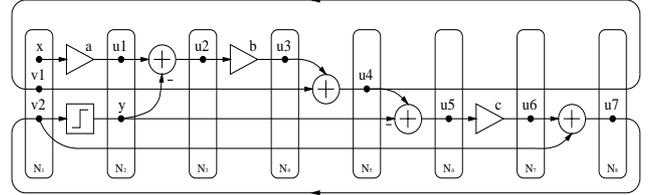


Figure 5

F. Stability of second order noise shaping loop

With modified scaling factor, the stability is proved based on the topology given above.

III. IMPLEMENTATION

A. Hardware Architecture

We implemented the basic second order noise shaping loop as our circuit for digital driving voice and audio load. The timing schedule is given in the above figure. In the following figure, illustrates the schematic view is given.

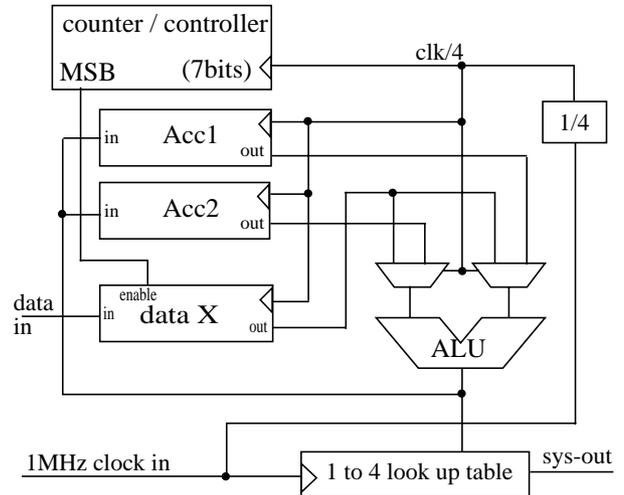


Figure 6

The system block diagram is given in the above figure. We decide the OSR is larger than 64, 256 is used to satisfy both the SNR and the power consumption. There are two clocks

running inside the system, the system clock is 2MHz. It drives a 1 - 4 look up table and convert 2 bits output from ALU to one bit sys-out. The data rate on the input of the table is 2-bit@0.5MHz. The data rate on the output of the table is 1bit@2MHz. The data rate of the system data in is 8k. The input data width is 16 bits.

When the clk/4 goes high the multiplexers will choose the X register and the Acc1 register to be inputs to the ALU. The ALU will then calculate the next value of the Acc1 register. This value will be loaded into the Acc1 register when clk/4 goes low. At the same time the multiplexers will choose the Acc1 register (containing the new value) and the Acc2 register as inputs to the ALU. The ALU will calculate the new value of the Acc2 register and with that also a new output value since MSB in the Acc2 is the output. This value will be loaded into the Acc2 register when the clk/4 goes high. Each time the clk/4 goes high the counter is incrementing and when it reaches 63 a new sample will be loaded into the X register.

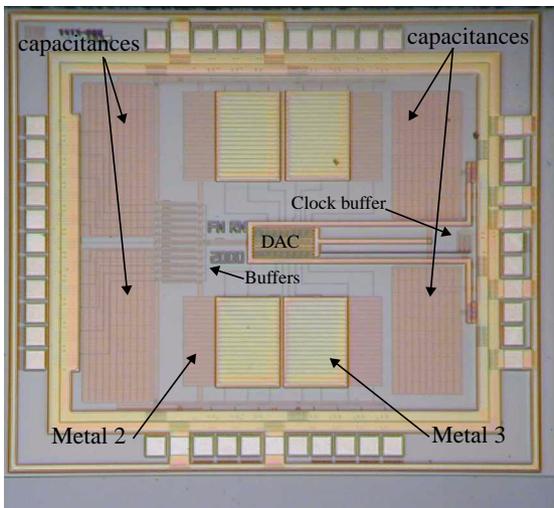


Figure 7

B. Finite State Machine

The simple finite state machine keeps counting based on the period of 64. The FSM is very simple due to the fact that the clock signal controls all building block except one, the X register. The only thing the FSM needs to do is to send a load pulse to the X register every 64th clk/4 cycle when the reset signal is inactivated. When the reset signal is activated it should load the register ones every clock cycle. To accomplish this a simple 6 bit counter and a flip-flop was used.

C. Chip design

The technology used in this design was a 0.35 μ m process from AMS. The process has got three different metal layers and two polysilicon layers.

Special pads for inductive load is considered at the beginning of the research. We feel that it was not necessary after we simulated the driving for different voice and audio load. On the test chip, normal driving pads with 10mA driving capacity are used. The number of required pads related to load impedances are also simulated. The measurement on the test chip gives perfect results matching what we got from simulation.

IV. MEASUREMENT

The average power consumption is 0.056mW when power supply is 2.5V at 2MHz clock (oversampling) frequency. The SRN is larger than 80dB when a mono frequency full swing signal was applied as the stimuli.

V. CONCLUSIONS

A 256 times oversampling second-order noise shaping loop has been verified with behavioral models in Matlab and Fdlab. The designed oversampling NSL is implemented in silicon as the digital circuit directly driving the voice and audio load. A chip has been made that meets the power requirements. The size of the core is about the size of two normal pad sizes. (about 0.02mm²) which is much smaller than any kind of DA converter based on analog circuit technique.

The circuit from the paper gives the design of digital driving voice and audio load for any CMOS silicon technology as a silicon independent IPR design.

ACKNOWLEDGMENTS

Authors would thank to Stig Stuns, Freehand DSP AB, Sundbyberg-Stockholm, Sweden for useful discussions.

REFERENCES

- [1] S. R. Norsworthy et al, (editor) Delta-Sigma Data Converters, IEEE press, ISBN 0-7803-1045-4
- [2] David A. Jones, Ken Martin, Analog Integrated Circuit Design
- [3] Lars Wanhammar, *DSP Integrated Circuits*. Academic press, 1999