Measurement: Sensors xxx (xxxx) xxx

Contents lists available at ScienceDirect

Measurement: Sensors

journal homepage: www.sciencedirect.com/journal/measurement-sensors

Digital to analog converter based on pulse-width adding

ARTICLE INFO

Keywords:
Digital-to-analog converter
DAC
Pulse width
PWM

ABSTRACT

This paper presents a digital-to-analog converter (DAC) based on the pulse width created naturally by the binary counting sequence. The concept involves combining the counting impulses of the bits to form a pulse-width modulated signal, where the mean voltage is proportional to the input digital code. We propose a circuit capable of achieving this using general-purpose components. Although our design targets 8 bits, it is scalable for any number of nibbles.

The paper details simulations conducted to verify the proper functioning of the circuit and to evaluate its performance. Tests were performed to determine the static characteristics of the converter, measure its differential nonlinearity (DNL), and observe its step response.

In static terms, the converter exhibited negligible gain and offset errors, with a DNL below 1 LSB (least significant bit). The converter operated correctly without any missing codes.

In dynamic terms, the converter demonstrated a bandwidth of 10 kHz and behaved like a second-order low-pass filter with critical damping.

1. Introduction

A digital-to-analog converter (DAC) is a circuit that transforms a digital code (a set of n bits) into an analog quantity, usually an electrical voltage. DAC circuits are used in various applications, including analog-to-digital conversion, digitally controlled signal generators, and automatic control systems, among others [1–4]. Fig. 1 shows the DAC's electrical symbol, including supply voltages (V_{DD} and V_{SS}) and the reference voltage input (V_{REF}), which serves as a reference for the converter's operation.

The two most common architectures for DACs are the R-2R ladder network and the delta-sigma ($\Delta\Sigma$) structure. The following paragraphs review these architectures to better contextualize our work.

1.1. R-2R resistor ladder network

The R-2R ladder network is probably the most popular structure for DACs [5,6]. Fig. 2 presents the overall view of the R-2R network, and Fig. 3 shows the details of a single node.

Each node in the R-2R network sees to its right an equivalent resistance whose value is always 2R. Therefore, the current that reaches the node is divided in half, with one half going down to the electronic switch, and the other half continuing downstream where it will be divided in half again. The bits of the input digital code control the switches so that each bit contributes twice the current of the previous bit, from the most significant bit (MSB) on the left to the least significant bit (LSB) on the right. The currents from all bits are summed to produce a total current proportional to the input digital code:

$$I = \frac{V_{REF}}{2R} \times a_{n-1} + \frac{V_{REF}}{4R} \times a_{n-2} + \ldots + \frac{V_{REF}}{2^n R} \times a_o \Leftrightarrow$$

$$I = \frac{I_{REF}}{2^n} \times \left(2^{n-1} a_{n-1} + 2^{n-2} a_{n-2} + \ldots + 2^0 a_0\right) \Leftrightarrow$$

$$I = \frac{I_{REF}}{2^n} \times D \tag{1}$$

where $I_{REF} = \frac{V_{REF}}{R}$ is the current supplied by the reference voltage, and D is the decimal value of the input digital code. Finally, the current is converted to a voltage by the operational amplifier, resulting io an output voltage given by:

$$V_O = -R \times I = -\frac{V_{REF}}{2^n} \times D \tag{2}$$

1.2. Delta-sigma ($\Delta\Sigma$) structure

The delta-sigma ($\Delta\Sigma$) structure [7–9] has gained popularity in recent years. It employs oversampling techniques to achieve high resolution, albeit at the expense of increased latency and reduced bandwidth.

Fig. 4 shows the working diagram of a delta-sigma ($\Delta\Sigma$) DA converter. The input is a digital code (represented by the thickest line), and the output is an analog voltage. For explanatory purposes, we will assume a signed representation is subtracted from a full-scale positive or negative number using the same number of bits. For example, an 8-bit incoming digital code, ranging from −128 to +127 as a signed

https://doi.org/10.1016/j.measen.2024.101418

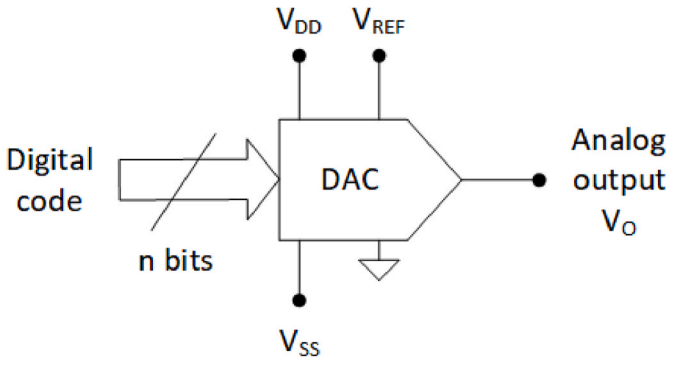

Fig. 1. Electrical symbol of a DAC.

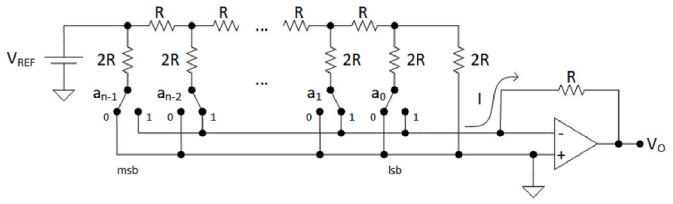

Fig. 2. R-2R ladder network.

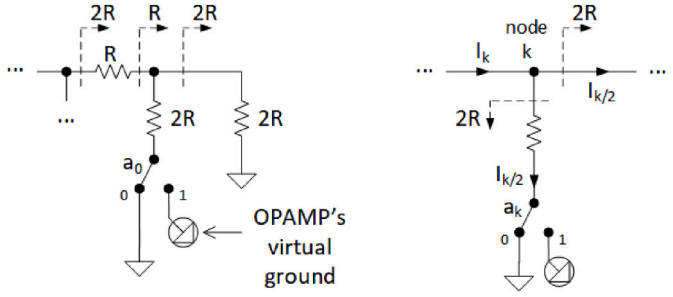

Fig. 3. Detail of single node of the R-2R ladder network.

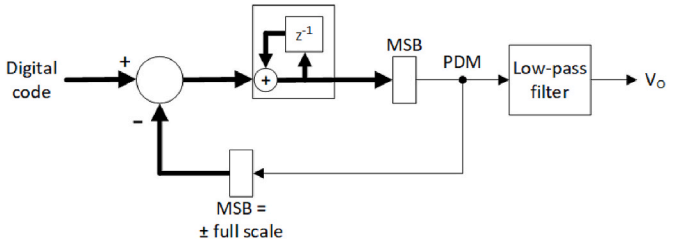

Fig. 4. $\Delta\Sigma$ DAC.

integer, would be subtracted from either -128 or $+127$, depending on the state of the feedback.

The difference is applied to the input of an accumulator (the z^{-1} block). If the difference is a positive, the output of the accumulator ramps up; if the difference is a negative, the output of the accumulator ramps down. The MSB block selects the most significant bit of the word (the sign bit) and toggles the feedback state (and the output) whenever the accumulator causes the sign to change. This results in a pulse-density-modulated (PDM) output waveform, whose mean value is proportional to the input digital code. Averaging can be achieved with a simple low-pass filter.

The main contribution of this paper is the proposal of a new type of DA converter that leverages the pulse width created naturally by the binary counting sequence. Our approach is likely closer to the $\Delta\Sigma$ structure, as it does not require a reference voltage and takes several clock cycles to complete a conversion. Additionally, it is simpler to understand and easier to implement, making it a valuable contribution to the state of the art.

The paper is organized as follows: Section 2 presents the new DA circuit and explains its operation. Section 3 describes the experimental tests conducted to validate our prototype and discusses the results obtained. Finally, Section 4 summarizes the main conclusions of the work.

2. Methods and procedures

The proposed circuit, shown in Fig. 5, takes advantage of the pulse width created naturally by the binary counting sequence. The circuit is based on nibbles because it uses a 4-bit binary counter (4516).

If we examine the binary counting sequence shown in Fig. 6, we observe that the width of the pulses varies according to the bit's weight. In general, all bits have a pulse width of 50 %, but this value is achieved through narrower pulses for the least significant bits and wider pulses for the more significant bits. The idea is to filter the first pulse of each bit (only the first occurrence) and combine them into a single signal. The result will be a binary-weighted pulse that can be used for digital-to-analog conversion.

The filtering can be made by a simple logical circuit. For example, the narrower the pulse, corresponding to the LSB, will be filtered if the corresponding input bit (A_0) is true AND the output of the binary counter is $Q_3Q_2Q_1Q_0 = 0001$. This leads to the logical circuit shown in Fig. 7a, where the signal X_0 pulses with a width of T_{clk} if $A_0 = 1$ (where T_{clk} is the clock period of the binary counter).

The same applies for the remaining bits:

- The $2T_{clk}$ pulse is filtered if the corresponding input bit (A_1) is true AND the output of the binary counter is $Q_3Q_2Q_1Q_0 = 001X$, leading to the circuit shown in Fig. 7b.
- The $4T_{clk}$ pulse is filtered if A_2 is true AND the output of the binary counter is $Q_3Q_2Q_1Q_0 = 01XX$, leading to the circuit shown in Fig. 7c.
- The $8T_{clk}$ pulse is filtered if A_3 is true AND the output of the binary counter is $Q_3Q_2Q_1Q_0 = 1XXX$, leading to the circuit shown in Fig. 7d.

In the end, we combine all the filtered pulses with a OR gate, leading to the output line *PWM* shown in Fig. 7e. After some optimizations to reduce the number of gates, we arrive at the circuit shown in Fig. 8. This circuit is contained within the block labelled "Pulse Filter" in Fig. 5.

For an 8-bit converter, we need to combine the outputs of two pulse filters, one for each nibble. This is achieved through the summing stage *U*3, which assigns a unitary weight to the most significant nibble and a weight of 1/16 to the least significant nibble. Additionally, the stage incorporates a first-order low-pass filter with a cutoff frequency of $\frac{1}{2\pi R_2 C_2}$ Hz, used to extract the average voltage from the pulses. A final stage (*U*4) can be used to apply further filtering and get a positive output voltage.

The pulse filters take 16 clock periods to complete their work. With a clock frequency of 4 MHz on the main counter, they generate 250 kHz signals, which are filtered by R_2C_2 and R_5C_3 (10 kHz cutoff frequency) resulting in a smoothed analog voltage at the output. It should be emphasized that the pulse filters work in parallel, not sacrificing excessively the speed of operation of the circuit. The pair R_1C_1 generates a brief pulse that resets the counter at power-up.

2.1. Example of operation

Let's examine how the circuit functions by considering an example.

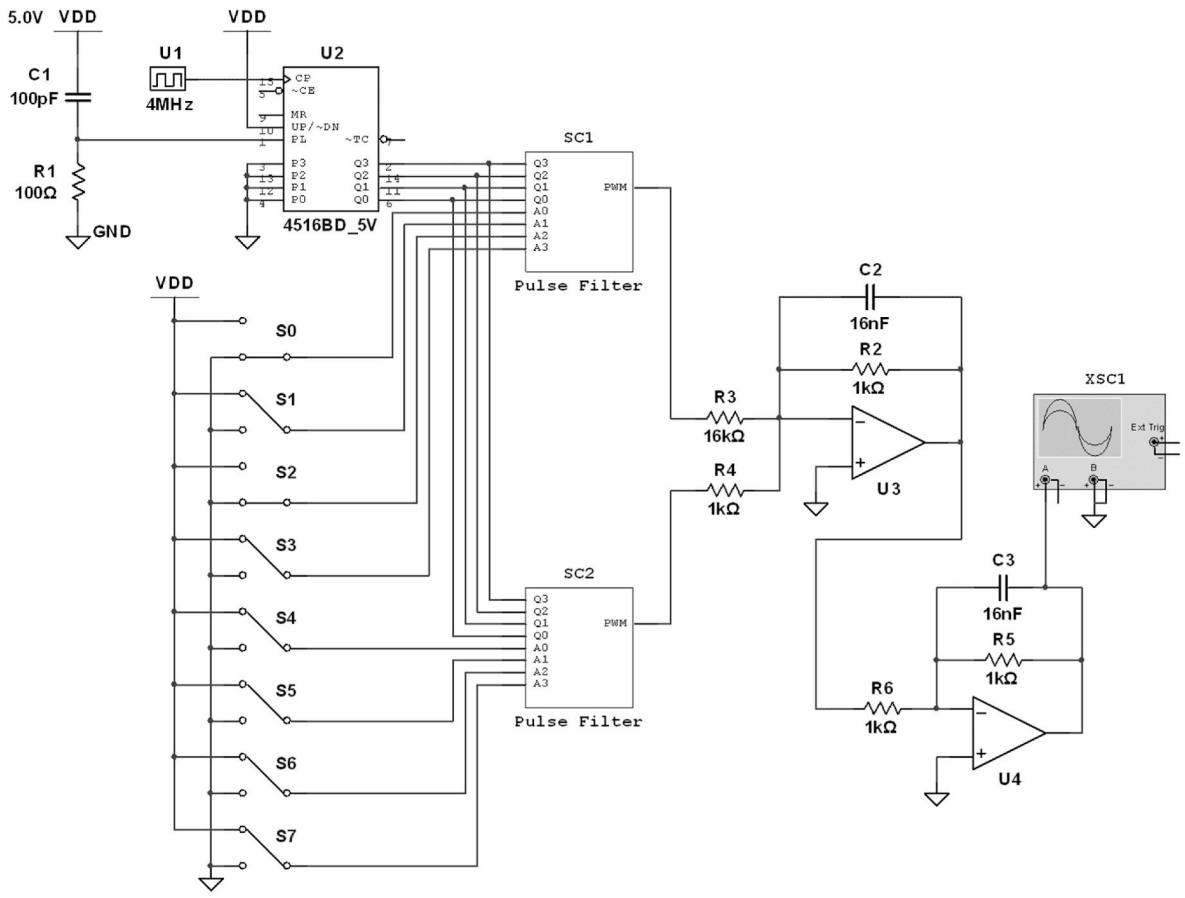

Fig. 5. 8-bit DAC based on pulse-width adding. The input digital word is $S7 - S0$, with $S7$ being the most significant bit (MSB) and $S0$ the least significant bit (LSB). The analog output is taken at the output of $U4$.

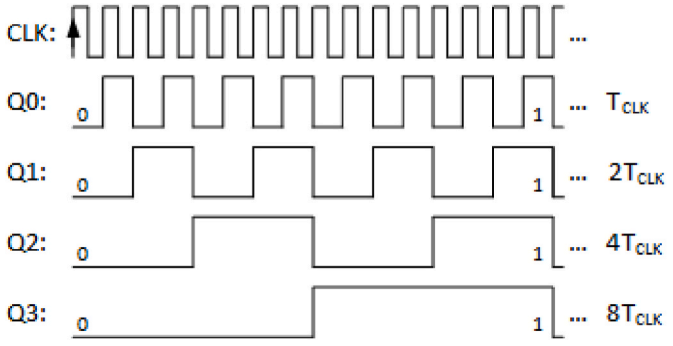

Fig. 6. Output of the binary counter. The width of the pulses increases as a power of 2.

Suppose the circuit in Fig. 5 receives the input digital code 0100 0010. The switches $S6$ (MSB) and $S1$ will be connected to V_{DD}, while all the other switches will be grounded. The pulse filter $SC2$ will process the nibble $A_3A_2A_1A_0 = 0100$, and the pulse filter $SC1$ will process the nibble $A_3A_2A_1A_0 = 0010$.

Inside the pulse filter $SC2$, the signals $X3$, $X1$ and $X0$ will remain low because $A_3 = A_1 = A_0 = 0$. However, the signal X2 will pulse high when $Q_3 = 0$ and $Q_2 = 1$; in other words, it will contain the first hight pulse of Q_2 (only the first occurrence). Consequently, the output (PWM line) will be square wave with a pulse-width equal to $4T_{clk}$ and a mean voltage equal to $\frac{1}{4} \times V_{DD}$.

The same explanation applies to the pulse filter $SC1$, but in this case, only the first high pulse of Q_1 passes to the output. The result will be

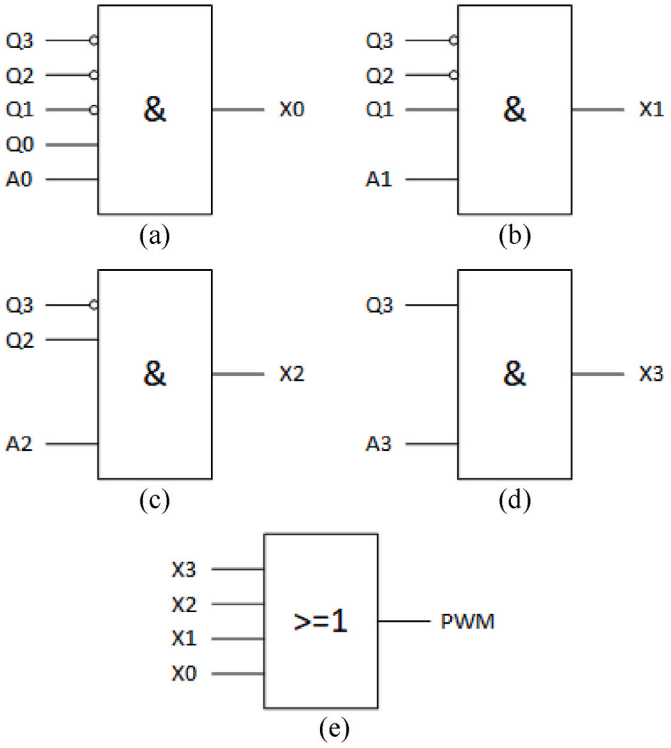

Fig. 7. Pulse filtering from the LSB (X_0) to the MSB (X_3) followed by adding (OR gate).

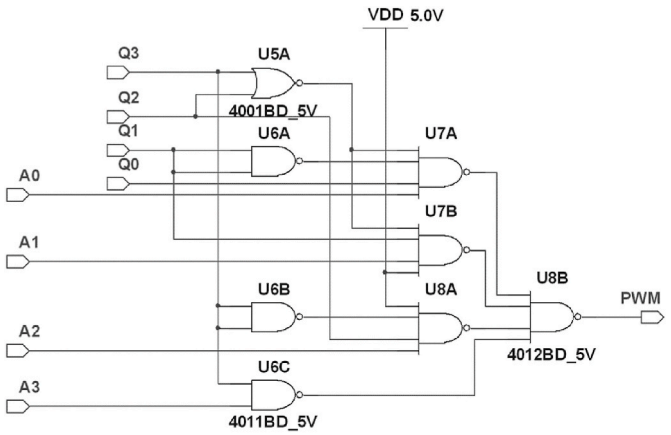

Fig. 8. Pulse filter sub-circuit.

square wave with a pulse-width equal to $2T_{clk}$ and a mean voltage equal to $\frac{1}{8} \times V_{DD}$.

The *U*3 stage sums the mean voltage of both pulse filters, giving a weight 1/16 to the least significant nibble. The final output will be an analog voltage equal to $\left(\frac{1}{4} \times V_{DD}\right) + \left(\frac{1}{16} \times \frac{1}{8} \times V_{DD}\right)$, which leads to 1289 mV assuming $V_{DD} = 5\ V$.

3. Results and discussion

This section describes the simulations carried out to verify the proper functioning of the circuit and evaluate its performance. The simulations were done using NI Multisim v14.3 running on a computer characterized by Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz, 16 GB RAM, and Windows 11 Home 64 bits. The simulations included the following tests:

- Static characteristic (output voltage versus input digital code in steady state).
- Differential nonlinearity (DNL).
- Step response.

Each of these tests will be explained in more detail in a dedicated section.

3.1. Static characteristic

The static characteristic of the DA converter was obtained by applying all the binary combinations at the input and recording the corresponding voltages at the output, once they had stabilized.

Table 1 shows the mean output voltages obtained for several input binary combinations (all zeros, all ones, and powers of 2). Fig. 9 shows the plot of the static characteristic, with the horizontal axis representing the input digital code (integer value) and the vertical axis representing the mean output voltage. The characteristic is well adapted (R = 1) by a straight line with slope equal to 19.531 mV/LSB and an offset equal to 84.2 μV. Looking at these values, we see that the slope matches the theoretical value (5/256) and that the offset error is negligible (0.0043 LSB).

We also measured the RMS value of the output voltage to have an idea of the noise generated by the circuit (see the last column of Table 1 and Fig. 10).

The noise level at the output is related to the number of edges of the *PWM* signals. For low values of the digital code, the *PWM* signal of the most significant nibble rests at 0 V most of the time. Conversely, for high values of the digital code, it rests at 5 V most of the time. For intermediate values, it oscillates between 0 and 5 V thus generating more noise. This explains the hump observed in Fig. 10. It is important to note that

Table 1
Key values of the static characteristic.

Input binary combination	Mean output voltage (mV)	RMS output voltage (mV)
0000 0000	0	0
0000 0001	19.531	0.054
0000 0010	39.062	0.096
0000 0100	78.123	0.155
0000 1000	156.246	0.218
0001 0000	312.494	0.869
0010 0000	624.987	1.538
0100 0000	1250	2.484
1000 0000	2500	3.493
1111 1111	4980	0.835

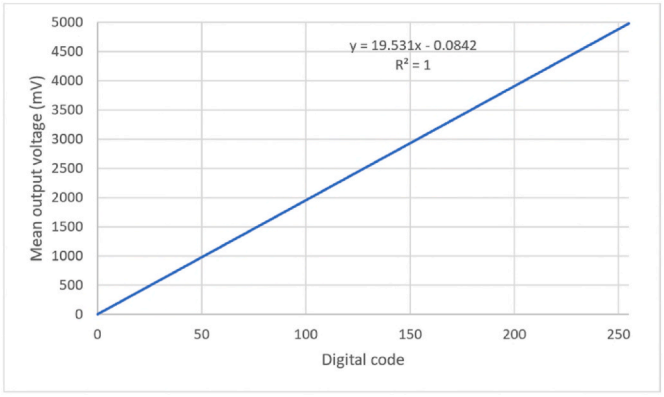

Fig. 9. Static characteristic.

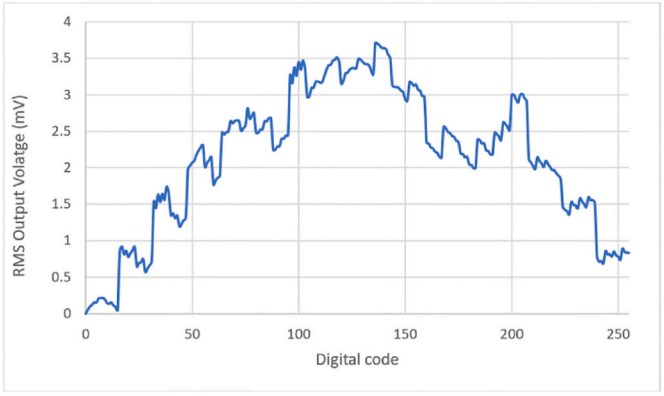

Fig. 10. Noise figure.

the maximum value of noise is 3.5 mV RMS, which is much smaller that the LSB.

3.2. Differential nonlinearity (DNL)

The DNL measures the difference between the measured and the ideal output voltages for successive digital codes [10]:

$$DNL(i) = \frac{V_{out}(i+1) - V_{out}(i)}{\text{Ideal } LSB} - 1 \tag{3}$$

The DNL thus defined is expressed as a fraction of the LSB, but it can also be expressed as a voltage or a percentage of full scale.

In the present case, our converter's DNL was obtained from the values of the static characteristic (considering mean output voltages in steady state). The result, shown in Fig. 11, uses the definition of equation (3).

We observe that the DNL is well below 1 LSB, meaning that the converter has no missing codes.

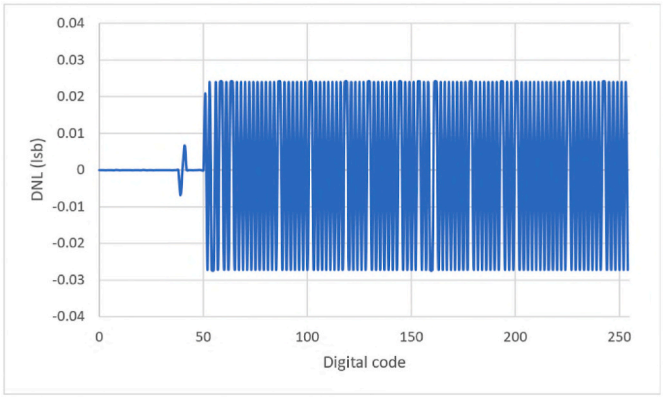

Fig. 11. DNL.

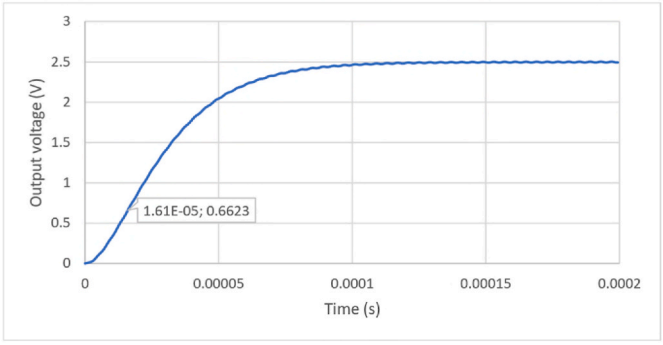

Fig. 12. Step response.

3.3. Step response

The step response is a valuable method for identifying the dynamic characteristics of a system [11]. Following this approach, we applied a sharp change to the input, shifting from code 0000 0000 to code 1000 0000, and recorded the output voltage over time. The result is shown in Fig. 12.

We observe that the dynamics of the converter are dominated by the second-order low-pass filter formed by $U3$ and $U4$. The converter exhibits the same damping as the filter, which is precisely equal to one (critical damping), corresponding to a double real pole.

Let us consider that the converter is modelled by the transfer function $G(s) = \frac{G_0}{(\tau s+1)^2}$, where G_0 represents the stationary gain of the converter, and τ represents the time constant of its double real pole. Such system has a unit step response given by:

$$y(t) = G_0\left[1-\left(1+\frac{t}{\tau}\right)e^{-\frac{t}{\tau}}\right]u(t) \tag{4}$$

where $u(t)$ represents the unit step function.

At time $t = \tau$ the response is equal to 26.4% of the final value; in other words, the point where the output voltage reaches 26.4% corresponds to time constant τ. That point is highlighted in Fig. 12, leading to a time constant $\tau \approx 16\ \mu s$, which corresponds to a double pole at 62500 rad/s and a cutoff frequency around 10 kHz (with −6 dB loss). These are the values for which the second-order low-pass filter (around $U3$ and $U4$) was designed.

If we look again to the circuit of Fig. 12, we see that the main clock frequency (4 MHz) is successively divided 2, 4, 8 and 16 by the binary counter, reaching a minimum value of 250 kHz. In other words, the *PWM* signals have frequencies of 250 kHz and above. By designing the second-order low-pass filter with a cutoff frequency of 10 kHz, two orders of magnitude lower, we ensure that the output voltage is smoothed sufficiently to not compromise the performance of the converter, as confirmed by the simulations. If we want to increase the bandwidth of the converter, we will need to increase the frequency of the main clock proportionally.

4. Conclusions

The paper presented a DA converter that leverages the pulse width naturally created by the binary counting sequence. The core idea is to combine the counting pulses of the bits to form a pulse-width-modulated signal, whose mean voltage is proportional to the input digital code. We introduced a circuit capable of performing this task using general-purpose components. While the circuit was designed for 8 bits, it is scalable to accommodate any number of nibbles. However, it is important to note that increasing the number of nibbles will result in longer processing times and reduced bandwidth.

The converter was tested to ensure proper functionality and to evaluate its performance. Tests were conducted to determine the static characteristics of the converter and to measure its step response.

In static terms, the converter was characterized by negligible gain and offset errors, and a DNL below 1 LSB. The converter worked properly without missing codes.

In dynamic terms, the converter was characterized by a bandwidth of 10 kHz, and a behaviour of a second-order low-pass filter with critical damping. The dynamic behaviour of the circuit is mostly determined by the active filters at the output.

In the future, we intend to work at a lower level by implementing the digital part of the circuit using MOSFET transistors to save hardware and increase velocity. We are also thinking in replacing the analog part (the two active filters) by a purely digital output based on pulse width.

Acknowledgments

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

References

[1] Bill McCulley, Bridging the divide: a DAC applications tutorial, Texas Instru. (2010). Online, https://www.ti.com/lit/an/snaa129/snaa129.pdf. (Accessed 20 April 2024).
[2] R.H. Walden, Analog-to-digital converter survey and analysis, IEEE J. Sel. Area. Commun. 17 (4) (April 1999) 539–550.
[3] L.F. Rahman, F.A. Rudham, M.B.I. Reaz, M. Marufuzzaman, The evolution of digital to analog converter, in: International Conference on Advances in Electrical, Electronic and Systems Engineering (ICAEES), Putrajaya, Malaysia, 2016, pp. 151–154.
[4] A. Cruz Serra, P. Daponte, L. Michaeli, ADC and DAC modelling and testing – state of the art, IMEKO TC4 Workshop 2005, Poland, Online, https://www.imeko.org/publications/tc4-2005/IMEKO-TC4-2005-001.pdf, 2005. (Accessed 20 April 2024).
[5] Walt Kester, Basic DAC architectures II: binary DACs, Analog Devices (2009). Online, https://www.analog.com/media/en/training-seminars/tutorials/MT-015.pdf. (Accessed 20 April 2024).
[6] TT electronics. R/2R ladder networks, Application note. Online [accessed 20 April 2024] https://www.digikey.com/Site/Global/Layouts/DownloadPdf.ashx?pdfUrl=EB5EB1AE94174E479C9188D1EF66BA24.
[7] Analog Devices. Sigma-delta ADCs and DACs, Application note (AN-283). Online [accessed 20 April 2024] https://www.analog.com/media/en/technical-documentation/application-notes/29252429152571724505492368045817 1an283.pdf.
[8] Richard Schreier, Shanthi Pavan, Gabor C. Temes, Understanding Delta-sigma Data Converters, Wiley, March 2017. ISBN:9781119258278.

[9] M.S. Enuchenko, A.S. Korotkov, Digital-to-analog converters based on delta-sigma modulators, J. Commun. Technol. Electron. 67 (April 2022) 1–16.
[10] Texas Instruments. Understanding data converters, Application report, 1999. Online [accessed 18 April 2024] https://www.ti.com/lit/an/slaa013/slaa013.pdf.
[11] Tao Liu, Qing-Guo Wang, Hsiao-Ping Huang, A tutorial review on process identification from step or relay feedback test, J. Process Control 23 (10) (2013) 1597–1623.

Vítor Viegas[a,c,*], J.M. Dias Pereira[b,c]
[a] CINAV – Centro de Investigação Naval, Escola Naval, Almada, Portugal
[b] Escola Superior de Tecnologia de Setúbal, Instituto Politécnico de Setúbal, Setúbal, Portugal
[c] IT – Instituto de Telecomunicações, Lisboa, Portugal

[*] Corresponding author. CINAV – Centro de Investigação Naval, Escola Naval, Almada, Portugal.
E-mail addresses: vitor.viegas@escolanaval.pt (V. Viegas), dias.pereira@estsetubal.ips.pt (J.M.D. Pereira).